



UNIVERSIDADE FEDERAL DO CEARÁ  
DEPARTAMENTO DE ENGENHARIA DE TELEINFORMÁTICA  
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE TELEINFORMÁTICA

# **Métodos Tensoriais para Estimação Cega de Assinaturas Espaciais**

*Dissertação de Mestrado*

**Paulo Ricardo Barboza Gomes**

FORTALEZA – CEARÁ  
MARÇO 2014



UNIVERSIDADE FEDERAL DO CEARÁ  
DEPARTAMENTO DE ENGENHARIA DE TELEINFORMÁTICA  
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE TELEINFORMÁTICA

# **Métodos Tensoriais para Estimação Cega de Assinaturas Espaciais**

**Autor**

**Paulo Ricardo Barboza Gomes**

**Orientador**

Prof. Dr. André Lima Férrer de Almeida

*Dissertação apresentada à  
Coordenação do Programa de  
Pós-graduação em Engenharia de  
Teleinformática da Universidade  
Federal do Ceará como parte dos  
requisitos para obtenção do grau  
de **Mestre em Engenharia de  
Teleinformática.***

FORTALEZA – CEARÁ

MARÇO 2014

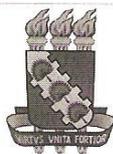
Dados Internacionais de Catalogação na Publicação  
Universidade Federal do Ceará  
Biblioteca de Pós-Graduação em Engenharia - BPGE

---

G616m      Gomes, Paulo Ricardo Barboza.  
2014        Métodos tensoriais para estimação cega de assinaturas espaciais / Paulo Ricardo Barboza Gomes. –  
84 f. : il. color., enc. ; 30 cm.

Dissertação (mestrado) – Universidade Federal do Ceará, Centro de Tecnologia, Departamento de Engenharia de Teleinformática, Programa de Pós-Graduação em Engenharia de Teleinformática, Fortaleza, 2014.  
Concentração: Sinais e Sistemas.  
Orientação: Prof. Dr. André Lima Férrer de Almeida.

1. Teleinformática. 2. Processamento de sinais. 3. Assinatura espacial. I. Título.



UNIVERSIDADE FEDERAL DO CEARÁ  
CENTRO DE TECNOLOGIA  
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE TELEINFORMÁTICA  
CAMPUS DO PICI, CAIXA POSTAL 6007 CEP 60.738-640  
FORTALEZA - CEARÁ - BRASIL  
FONE (+55) 85 3366-9467 - FAX (+55) 85 3366-9468

**PAULO RICARDO BARBOZA GOMES**

**MÉTODOS TENSORIAIS PARA ESTIMAÇÃO CEGA DE ASSINATURAS  
ESPACIAIS**

Dissertação submetida à Coordenação do Programa de Pós-Graduação em Engenharia de Teleinformática, da Universidade Federal do Ceará, como requisito parcial para a obtenção do grau de Mestre em Engenharia de Teleinformática.  
Área de concentração: Sinais e Sistemas.

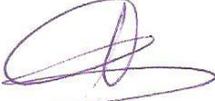
Aprovada em: 14/03/2014.

**BANCA EXAMINADORA**

  
Prof. Dr. André Lima Férrer de Almeida (Orientador)  
Universidade Federal do Ceará

  
Prof. Dr. Charles Casimiro Cavalcante  
Universidade Federal do Ceará

  
Prof. Dr. João Paulo Carvalho Lustosa da Costa  
Universidade de Brasília

  
Prof. Dr. Gérard Favier  
Université de Nice Sophia Antipolis

# Resumo

Nesta dissertação o problema de estimação de assinaturas espaciais e consequentemente da direção de chegada dos sinais incidentes em arranjos Linear, *2L-Shape* e Planar é investigado. Métodos baseados em decomposições tensoriais são propostos para tratar o problema de estimação cega de assinaturas espaciais desconsiderando a utilização de sequências de treinamento e o conhecimento da estrutura de covariância das fontes. Ao assumir que a potência das fontes varia entre blocos de tempos sucessivos, decomposições para tensores de terceira e quarta ordem obtidas a partir da covariância espacial e espaço-temporal dos dados recebidos no arranjo de sensores são propostas, a partir das quais algoritmos iterativos são formulados para estimar a assinatura espacial das fontes. Em seguida, uma maior diversidade espacial é alcançada utilizando a técnica *Spatial Smoothing* na recepção de sinais nos arranjos *2L-Shape* e Planar. Nesse caso, as estimações da direção de chegada das fontes não podem ser obtidas diretamente a partir dos algoritmos formulados de forma que a fatoração do produto de Khatri-Rao é incorporada a estes algoritmos, tornando possível a obtenção de estimações para os ângulos de azimute e elevação a partir das matrizes obtidas utilizando este método. Uma característica marcante dos métodos tensoriais propostos está presente na eficiência obtida no tratamento de casos em que a matriz de covariância das fontes é não-diagonal e desconhecida, o que geralmente ocorre quando se trabalha com covariâncias de amostras reais calculadas a partir de um número reduzido de *snapshots*.

**Palavras-Chave:** Processamento de sinais em arranjos, estimação de assinatura espacial, decomposições tensoriais.

# Abstract

In this dissertation, the problem of spatial signature and direction of arrival estimation in Linear, 2L-Shape and Planar arrays is investigated. Methods based on tensor decompositions are proposed to treat the problem of estimating blind spatial signatures disregarding the use of training sequences and knowledge of the covariance structure of the sources. By assuming that the power of the sources varies between successive time blocks, decompositions for tensors of third and fourth orders obtained from spatial and spatio-temporal covariance of the received data in the array are proposed, from which iterative algorithms are formulated to estimate spatial signatures of the sources. Then greater spatial diversity is achieved by using the Spatial Smoothing in the 2L-Shape and Planar arrays. In that case, the estimation of the direction of arrival of the sources can not be obtained directly from the formulated algorithms. The factorization of the Khatri-Rao product is then incorporated into these algorithms, making it possible extracting estimates for the azimuth and elevation angles from matrices obtained using this method. A distinguishing feature of the proposed tensor methods is their efficiency to treat the cases where the covariance matrix of the sources is non-diagonal and unknown, which generally happens when working with sample data covariances computed from a reduced number of snapshots.

**Key-words:** Array signal processing, spatial signature estimation, tensor decompositions.

... aos meus pais Nazaré e Paulo, meus irmãos Matheus e Ruan, minha namorada Flávia Ingrid e a todos os familiares e amigos.

# Agradecimentos

A Deus, pelas graças concedidas, força em todos os momentos e pelo cuidado com que sempre escolhe os meus caminhos.

Aos meus pais Maria de Nazaré e Paulo Gomes, meus irmãos Pedro Matheus e Paulo Ruan, e demais familiares pela força, apoio e carinho que foram de fundamental importância para a realização desse sonho.

A minha namorada Flávia Ingrid, amor e amiga, por ser minha companheira não apenas na estrada da vida, mas também na vida acadêmica. Seus incentivos, paciência, carinho e compreensão foram fundamentais a este trabalho. Amo-te.

Ao Prof. Dr. André Lima Férrer de Almeida, pela ótima orientação e acolhimento no decorrer desses dois anos de mestrado.

Aos professores Dr. Charles Casimiro Cavalcante, Dr. João Paulo Carvalho Lustosa da Costa e Dr. Gérard Favier que participaram da banca examinadora e contribuíram diretamente para a melhoria deste trabalho.

Aos meus amigos Albano Oliveira Nunes e Fernando Filho pelas inúmeras conversas, conselhos e ajudas.

Aos amigos Gilderlan Araújo, Emanuel Sena, Jordan Paiva, Igor Sousa e Ítalo Vitor. Nossas conversas foram fundamentais para a conclusão desta dissertação.

E por fim agradeço à FUNCAP (Fundação Cearense de Apoio ao Desenvolvimento Científico e Tecnológico) pelo apoio financeiro concedido.

# Sumário

---

<b>Lista de Figuras</b>	<b>vii</b>
<b>Lista de Tabelas</b>	<b>ix</b>
<b>1 Introdução</b>	<b>1</b>
<b>2 Revisão Bibliográfica</b>	<b>4</b>
2.1 Modelo de Recepção de Sinais: Arranjo <i>2L-Shape</i> . . . . .	4
2.2 Modelo de Recepção de Sinais: Arranjo Planar . . . . .	8
2.3 Métodos Matriciais para Estimação de DoA . . . . .	10
2.3.1 Método MUSIC ( <i>Multiple Signal Classification</i> ) . . . . .	11
2.3.2 Método ESPRIT ( <i>Estimation of Signal Parameters by Rotational Invariance Techniques</i> ) . . . . .	16
2.4 Fundamentos de Álgebra Multilinear e Decomposições Tensoriais	21
2.4.1 Conceitos Básicos . . . . .	21
2.4.2 Decomposições Tensoriais . . . . .	23
2.5 Estimação de DoA Utilizando Decomposições Tensoriais . . . . .	30
<b>3 Métodos Tensoriais Propostos</b>	<b>32</b>
3.1 Receptor PARAFAC . . . . .	33
3.2 Receptor PARATUCK2 . . . . .	37
3.3 Receptor Tucker4 . . . . .	41
3.4 Condições de Identificabilidade . . . . .	45
3.5 Estimação da DoA a partir de $\hat{A}_{(final)}$ . . . . .	47
3.6 Arranjo <i>2L-Shape</i> com <i>Spatial Smoothing</i> . . . . .	48
3.7 Arranjo Planar sem <i>Spatial Smoothing</i> . . . . .	52
3.8 Arranjo Planar com <i>Spatial Smoothing</i> . . . . .	54
3.9 Least-Squares Khatri-Rao Factorization (LS-KRF) . . . . .	57
<b>4 Resultados Numéricos e Discussões</b>	<b>60</b>
4.1 Resultados para o Arranjo Linear Uniforme . . . . .	60
4.2 Resultados para o Arranjo <i>2L-Shape</i> . . . . .	66
4.3 Resultados para o Arranjo Planar . . . . .	70
4.4 Comparação entre ULA, <i>2L-Shape</i> e Planar . . . . .	73
<b>5 Conclusões e Perspectivas</b>	<b>76</b>



# Lista de Figuras

2.1	Arranjo <i>2L-Shape</i> com $N_x + N_y + N_z - 2$ sensores. . . . .	5
2.2	Arranjo Linear Uniforme com $N_z$ sensores. . . . .	7
2.3	Arranjo Planar com $N_x \cdot N_z$ sensores. . . . .	9
2.4	ULA com $N_z = 6$ sensores. . . . .	13
2.5	ULA com $N_z = 10$ sensores. . . . .	13
2.6	Arranjo Planar com $N_x = 4$ , $N_z = 4$ e SNR = 10dB. . . . .	15
2.7	Arranjo Planar com $N_x = 4$ , $N_z = 4$ e SNR = 25dB. . . . .	15
2.8	Divisão do Arranjo Planar em Sub-Arranjos. . . . .	20
2.9	Matriciação de um Tensor de Terceira Ordem. . . . .	22
2.10	Decomposição PARAFAC de um Tensor de Terceira Ordem. . . . .	24
2.11	Decomposição Tucker3. . . . .	27
2.12	Decomposição PARATUCK2. . . . .	29
3.1	Estrutura de Transmissão dos Dados. . . . .	33
3.2	Erro de Aproximação vs. <i>Snapshots</i> por Sub-Bloco. . . . .	37
3.3	<i>Spatial Smoothing</i> para um Arranjo Arbitrário. . . . .	48
3.4	Divisão do Arranjo <i>2L-Shape</i> em Sub-Arranjos. . . . .	49
3.5	<i>Spatial Smoothing</i> para o Arranjo Planar. . . . .	54
4.1	RMSE <i>versus</i> $T_s$ para $M = 2$ , $N = 5$ e $P = 20$ para SNR = 15 dB e configuração ULA no receptor. . . . .	61
4.2	Tempo médio de processamento <i>versus</i> $T_s$ para $M = 2$ , $N = 5$ e $P = 20$ para SNR = 15 dB e configuração ULA no receptor. . . . .	61
4.3	NMSE de $\hat{A}_{(final)}$ <i>versus</i> SNR para $M = 4$ , $N = 3$ e $P = 25$ considerando diferentes valores de $T_s$ e configuração ULA no receptor. . . . .	62
4.4	NMSE de $\hat{A}_{(final)}$ <i>versus</i> SNR para $N = 3$ , $T_s = 50$ e $P = 25$ considerando diferentes valores de $M$ e configuração ULA no receptor. . . . .	63
4.5	Número médio de iterações <i>versus</i> SNR para $N = 3$ , $T_s = 50$ e $P = 25$ considerando diferentes valores de $M$ e configuração ULA no receptor. . . . .	64
4.6	RMSE <i>versus</i> SNR para $M = 2$ , $N = 6$ , $T_s = 40$ e $P = 20$ e configuração ULA no receptor. . . . .	64
4.7	RMSE <i>versus</i> SNR para $M = 2$ , $N = 4$ , $T_s = 40$ e $P = 20$ e configuração ULA no receptor. . . . .	65
4.8	Tempo médio de processamento <i>versus</i> SNR para $M = 2$ , $N = 6$ , $T_s = 40$ e $P = 20$ e configuração ULA no receptor. . . . .	66

4.9 RMSE <i>versus</i> SNR para $M = 3, P = 25, T_s = 40$ e $N_x, N_y$ e $N_z$ iguais a 4 e configuração <i>2L-Shape</i> no receptor. . . . .	67
4.10 Número médio de iterações <i>versus</i> SNR para $M = 3, P = 25, T_s = 40$ e $N_x, N_y$ e $N_z$ iguais a 4 e configuração <i>2L-Shape</i> no receptor. . . . .	68
4.11 Tempo médio de processamento <i>versus</i> SNR para $M = 3, P = 25, T_s = 40$ e $N_x, N_y$ e $N_z$ iguais a 4 e configuração <i>2L-Shape</i> no receptor. . . . .	68
4.12 RMSE <i>versus</i> SNR para $M = 2, P = 25, T_s = 40$ e $N_x, N_y$ e $N_z$ iguais a 3 e configuração <i>2L-Shape</i> no receptor. . . . .	69
4.13 Tempo médio de processamento <i>versus</i> SNR para $M = 2, P = 25, T_s = 40$ e $N_x, N_y$ e $N_z$ iguais a 3 e configuração <i>2L-Shape</i> no receptor. . . . .	70
4.14 RMSE <i>versus</i> SNR para $M = 2, P = 25, T_s = 50$ e $N_x$ e $N_z$ iguais a 4 e configuração Planar no receptor. . . . .	71
4.15 Número médio de iterações <i>versus</i> SNR para $M = 2, P = 25, T_s = 50$ e $N_x$ e $N_z$ iguais a 4 e configuração planar no receptor. . . . .	71
4.16 RMSE <i>versus</i> SNR para $M = 2, P = 25, T_s = 50$ e $N_x$ e $N_z$ iguais a 4 e configuração Planar sem <i>Spatial Smoothing</i> . . . . .	72
4.17 RMSE <i>versus</i> SNR para $M = 2, P = 25, T_s = 50$ e $N_x$ e $N_z$ iguais a 4 e configuração Planar com <i>Spatial Smoothing</i> . . . . .	72
4.18 Tempo médio de processamento <i>versus</i> SNR para $M = 2, P = 25, T_s = 50$ e $N_x$ e $N_z$ iguais a 4 e configuração Planar no receptor. . . . .	73
4.19 RMSE (elevação) <i>versus</i> SNR para $M = 2, P = 20, T_s = 50$ e arranjos compostos por 9 sensores e utilizando a decomposição PARATUCK2. . . . .	74
4.20 RMSE (elevação) <i>versus</i> SNR para $M = 2, P = 20, T_s = 50$ e arranjos compostos por 9 sensores e utilizando a decomposição Tucker4. . . . .	74
4.21 RMSE (total) <i>versus</i> SNR para $M = 2, P = 20$ e $T_s = 50$ e arranjos compostos por 9 sensores no total. . . . .	75

# Lista de Tabelas

3.1 Estrutura da Matriz $\hat{A}_{(\text{final})}$ . . . . .	57
--	----

# Lista de Acrônimos

<b>ALS</b>	<i>Alternating Least Squares</i>
<b>BPSK</b>	<i>Binary Phase-Shift Keying</i>
<b>CANDECOMP</b>	<i>Canonical Decomposition</i>
<b>DFT</b>	<i>Discrete Fourier Transform</i>
<b>DoA</b>	<i>Direction of Arrival</i>
<b>DS-CDMA</b>	<i>Direct-Sequence Code-Division Multiple Access</i>
<b>ESPRIT</b>	<i>Estimation of Signal Parameters by Rotational Invariance Techniques</i>
<b>EVD</b>	<i>EigenValue Decomposition</i>
<b>LS-KRF</b>	<i>Least Squares Khatri-Rao Factorization</i>
<b>MIMO</b>	<i>Multiple-Input Multiple-Output</i>
<b>MUSIC</b>	<i>Multiple Signal Classification</i>
<b>NMSE</b>	<i>Normalized Mean Square Error</i>
<b>OFDM</b>	<i>Orthogonal Frequency-Division Multiplexing</i>
<b>PARAFAC</b>	<i>Parallel Factor Analysis</i>
<b>QALS</b>	<i>Quadrilinear Alternating Least Squares</i>
<b>RMSE</b>	<i>Root Mean Square Error</i>
<b>SDMA</b>	<i>Spatial Division Multiple Access</i>
<b>SNR</b>	<i>Signal-to-Noise Ratio</i>
<b>SVD</b>	<i>Singular Value Decomposition</i>
<b>TALS</b>	<i>Trilinear Alternating Least Squares</i>
<b>ULA</b>	<i>Uniform Linear Array</i>

# Lista de Artigos

## ► CONGRESSO

*Um Método Tensorial Eficiente para Estimação Cega de Assinaturas Espaciais.* Publicado no XXXI Simpósio Brasileiro de Telecomunicações - SBrT2013, 1-4 de Setembro de 2013, Fortaleza, CE.

*Fourth-Order Tensor Method for Blind Spatial Signature Estimation.* Aceito no IEEE International Conference on Acoustics, Speech and Signal Processing- ICASSP2014, 1-4 de Maio de 2014, Florença, Itália.

# Notação

No decorrer deste trabalho a seguinte notação será utilizada:

$x$	Escalares
$\mathbf{x}$	Vetores
$\mathbf{X}$	Matrizes
$\mathcal{X}$	Tensores
$\mathbf{x} \cdot \mathbf{y}$	Produto interno entre $\mathbf{x}$ e $\mathbf{y}$
$\mathbf{x} \circ \mathbf{y}$	Produto externo entre $\mathbf{x}$ e $\mathbf{y}$
$\mathbb{C}$	Conjunto dos valores complexos
$\mathbb{C}^{I_1}$	Vetor de dimensão $I_1$ com valores complexos
$\mathbb{C}^{I_1 \times I_2}$	Matriz de dimensão $I_1 \times I_2$ com valores complexos
$\mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$	Tensor de dimensão $I_1 \times I_2 \times \dots \times I_N$ com valores complexos
$\mathbf{A}^*$	Conjugado de $\mathbf{A}$
$\mathbf{A}^T$	Transposto de $\mathbf{A}$
$\mathbf{A}^H$	Conjugado transposto de $\mathbf{A}$
$\mathbf{A}^\dagger$	Pseudo-inversa de Moore-Penrose de $\mathbf{A}$
$\ \cdot\ _F$	Norma de Frobenius da Matriz ou Tensor
$\mathbf{I}_N$	Matriz identidade de dimensão $N$
$x_{i_1, i_2}$	$(i_1, i_2)$ -ésimo elemento de $\mathbf{X} \in \mathbb{C}^{I_1 \times I_2}$
$x_{i_1, i_2, \dots, i_N}$	$(i_1, i_2, \dots, i_N)$ -ésimo elemento de $\mathcal{X} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$
$\mathbf{X}_{i_1 \cdot} (\mathbf{X}_{\cdot i_2})$	$i_1$ -ésima linha ( $i_2$ -ésima coluna) de $\mathbf{X} \in \mathbb{C}^{I_1 \times I_2}$
$\mathbf{X}_{i_1 \cdot} \in \mathbb{C}^{I_2 \times I_3}$	$i_1$ -ésimo <i>slice</i> modo-1 de $\mathcal{X}$
$\mathbf{X}_{\cdot i_2} \in \mathbb{C}^{I_3 \times I_1}$	$i_2$ -ésimo <i>slice</i> modo-2 de $\mathcal{X}$
$\mathbf{X}_{\cdot \cdot i_3} \in \mathbb{C}^{I_1 \times I_2}$	$i_3$ -ésimo <i>slice</i> modo-3 de $\mathcal{X}$
$\times_n$	Produto modo- $n$
$[\mathcal{X}]_{(n)}$	Matriciação modo- $n$ de $\mathcal{X}$
$\mathbf{A} \otimes \mathbf{B}$	Produto de Kronecker entre as matrizes $\mathbf{A}$ e $\mathbf{B}$
$\mathbf{A} \diamond \mathbf{B}$	Produto de Khatri-Rao entre as matrizes $\mathbf{A}$ e $\mathbf{B}$
$\mathbf{A} \odot \mathbf{B}$	Produto de Hadamard entre as matrizes $\mathbf{A}$ e $\mathbf{B}$
$\text{vec}(\mathbf{X})$	Operador vetorização. Converte $\mathbf{X}$ em um vetor coluna $\mathbf{x}$ empilhando suas colunas
$\text{unvec}(\mathbf{X})$	Operação inversa da vetorização. Converte o vetor $\mathbf{x}$ de volta para uma matriz
$\text{vecd}(\mathbf{X})$	Converte os elementos da diagonal principal de $\mathbf{X}$ num vetor
$D_j(\mathbf{A})$	Representa uma matriz diagonal construída a partir da $j$ -ésima linha de $\mathbf{A}$
$\text{diag}(\mathbf{x})$	Forma uma matriz diagonal a partir do vetor $\mathbf{x}$
$k_A$	$k$ -rank da matriz $\mathbf{A}$

## Introdução

A estimação da Direção de Chegada (**DoA**, do inglês *Direction of Arrival*) é um problema chave em sistemas que utilizam arranjos de sensores no receptor. No contexto das comunicações móveis, este parâmetro é utilizado na localização de fontes e possibilita o uso de técnicas de formatação de feixes, possuindo dessa forma, um papel importante no projeto de sistemas de comunicação **SDMA** (do inglês, *Spatial Division Multiple Access*) [1]. A literatura sobre métodos matriciais utilizados na estimação de DoA é abundante. Uma visão geral de métodos clássicos é dada em [2] e [3]. As soluções existentes podem ser classificadas de diferentes formas dependendo das suposições assumidas (i) o não conhecimento de sinais piloto, que originam técnicas cegas de estimação, (ii) a utilização de modelos paramétricos ou não paramétricos para as assinaturas espaciais, (iii) o uso de independência estatística, entre outras.

Neste contexto, os métodos de estimação cegos vêm ganhando bastante atenção nos últimos anos. Entretanto, a maioria destes métodos não utiliza a estrutura multidimensional dos dados recebidos, que pode levar em consideração várias dimensões como código, polarização, espaço, tempo e/ou frequência. Em virtude disso, a fim de lidar com esta natureza multidimensional, as decomposições tensoriais têm sido amplamente aplicadas em problemas na área de processamento de sinais em arranjos [4] e [5] assim como em problemas de comunicação [6], [7] e [8]. Em um cenário cego a realização do processamento de sinais utilizando tensores em vez de matrizes promove vantagens significativas, dentre as quais podemos citar a melhoria nas condições de identificabilidade que geralmente são oriundas das propriedades encontradas nas decomposições tensoriais.

No contexto da utilização de tensores no desenvolvimento de técnicas para estimação de assinaturas espaciais e DoA, métodos baseados na decomposição PARAFAC (do inglês, *Parallel Factor Analysis*) [9] foram

propostos em [4], [5] e [10]. O método proposto em [10] é de interesse particular, pois não requer nenhum conhecimento sobre o canal de propagação e estrutura do arranjo de sensores que compõe o receptor, proporcionando desempenhos comparáveis ou até melhores que os métodos matriciais. Porém, esse método apresenta algumas limitações em sua modelagem, sendo necessário garantir que a matriz de covariância dos sinais das fontes seja perfeitamente conhecida e diagonal. Na prática, este comportamento corresponde a uma descorrelação entre os sinais transmitidos pelas diferentes fontes assim como uma situação em que o número de amostras coletadas (*snapshots*) necessárias para o cálculo numérico da covariância é suficientemente grande. O objetivo deste trabalho é desenvolver métodos que possam fornecer estimações precisas de DoA sem levar em consideração tais suposições idealizadas encontradas em [10].

Este trabalho aborda a estimação da matriz de assinaturas espaciais e conseqüentemente de DoA dos sinais incidentes nos arranjos Linear, *2L-Shape* e Planar utilizando covariâncias espaciais e espaço-temporais dos sinais recebidos. Os métodos propostos assumem que a potência dos sinais variam entre blocos de tempo sucessivos de modo que as decomposições tensoriais PARATUCK2 para um tensor de terceira ordem e Tucker4 para um tensor de quarta ordem são formuladas a partir do cálculo das covariâncias espaciais e espaço-temporais, respectivamente. A partir dos modelos tensoriais obtidos, algoritmos iterativos baseados em ALS (do inglês, *Alternating Least Squares*) e adicionados do LS-KRF (do inglês, *Least Squares Khatri-Rao Factorization*) são formulados para realizar a estimação da matriz de assinaturas espaciais. Nestes modelos, em contraste com o método proposto em [10], a estimação pode ser realizada adequadamente quando a covariância das fontes possuir uma estrutura arbitrária (não-diagonal) e desconhecida, o que torna os métodos propostos de grande interesse em ambientes práticos onde um número finito de *snapshots* é considerado.

Este trabalho está organizado da seguinte forma:

**Capítulo 2** – Neste capítulo será apresentada a fundamentação teórica utilizada no decorrer do trabalho. Serão apresentados os modelos matemáticos que descrevem o processo de recepção de sinais nos arranjos *2L-Shape* e Planar assim como uma compilação sobre métodos matriciais para estimação de DoA, dando uma ênfase maior aos métodos MUSIC e ESPRIT. Em seguida, serão apresentados conceitos de álgebra multilinear que serão amplamente utilizados durante a descrição das

decomposições PARAFAC, Tucker e PARATUCK2.

**Capítulo 3** – Neste capítulo serão apresentados os métodos tensoriais propostos para estimação de DoA. A modelagem do receptor será realizada para os arranjos *2L-Shape* e Planar utilizando ou não a propriedade de invariância ao deslocamento existente nestas configurações. Os conceitos apresentados no Capítulo 2 serão utilizados no decorrer da descrição da recepção dos sinais no arranjo de sensores assim como na formulação dos algoritmos iterativos.

**Capítulo 4** – Neste capítulo os resultados numéricos obtidos através de simulações computacionais serão apresentados e discutidos a fim de validar as formulações desenvolvidas. Estes resultados promovem uma análise de desempenho entre os métodos propostos no Capítulo 3 e os demais métodos utilizados como parâmetros de comparação.

**Capítulo 5** – Neste capítulo o trabalho será finalizado. Nele serão expostas as conclusões obtidas e perspectivas para trabalhos futuros.

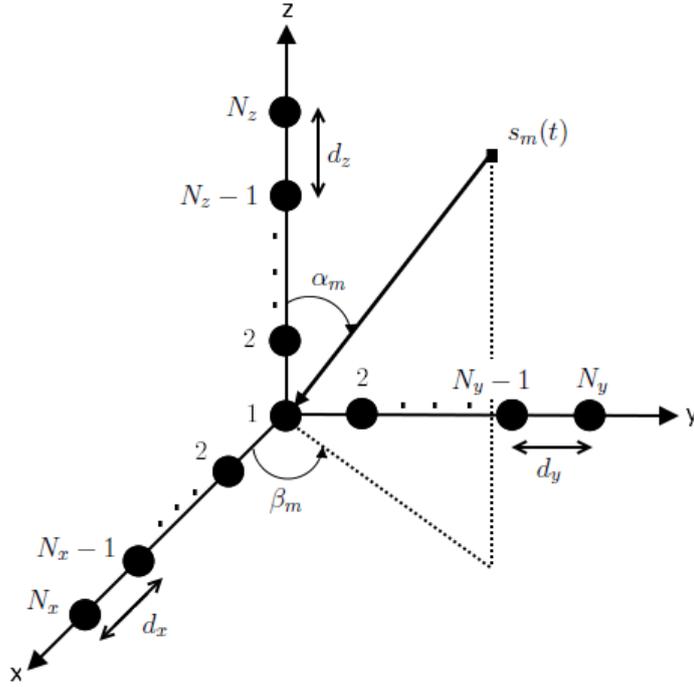
## Revisão Bibliográfica

Neste capítulo será realizada uma revisão bibliográfica dos conceitos fundamentais que servirão como base para o desenvolvimento deste trabalho. Inicialmente, os modelos de recepção e processamento de sinais em arranjos de sensores para as geometrias *2L-Shape* e Planar serão apresentados. Em seguida, será realizada uma breve revisão sobre métodos clássicos para estimação de DoA e os algoritmos MUSIC e ESPRIT, para os arranjos considerados, serão descritos detalhadamente. Por último, conceitos de álgebra multilinear e decomposições tensoriais serão apresentados e servirão de alicerce durante a descrição dos métodos de estimação de DoA propostos nesta dissertação os quais serão apresentados detalhadamente no Capítulo 3.

### 2.1 Modelo de Recepção de Sinais: Arranjo *2L-Shape*

Atualmente a estimação de DoA de múltiplos sinais que incidem simultaneamente em arranjos de sensores é de fundamental importância em diversas aplicações, principalmente nos casos que envolvem estimação conjunta dos ângulos de azimute e elevação dos sinais incidentes em arranjos 2D e 3D. Para esse tipo de problema diversos estudos abordando diferentes configurações de arranjos foram desenvolvidos, por exemplo, utilizando arranjos *L-Shape* [11], *V-Shape* [12] e *Z-Shape* [13]. Dentre as diferentes geometrias existentes uma das configurações de arranjos de sensores mais popular é a *2L-Shape* [14], [15]. Esta configuração é formada por três arranjos lineares posicionados nos eixos  $x$ ,  $y$  e  $z$  com sensores idênticos uniformemente espaçados de  $d_x$ ,  $d_y$  e  $d_z$  respectivamente. Cada arranjo é composto por  $N_x$ ,  $N_y$  e  $N_z$  sensores totalizando  $N_x + N_y + N_z - 2$  sensores distribuídos como ilustra a Figura 2.1.

Para descrever o modelo de sinal para o arranjo *2L-Shape* considere que a  $m$ -ésima fonte do sinal  $s_m(t)$  recebido no  $t$ -ésimo *snapshot* encontra-se suficientemente distante de forma que a onda eletromagnética recebida



**Figura 2.1:** Arranjo 2L-Shape com  $N_x + N_y + N_z - 2$  sensores.

seja compreendida como uma onda plana que incide no arranjo com um ângulo de elevação  $\alpha_m$ , ângulo de azimute  $\beta_m$  e vetor de onda  $\mathbf{k}_m = [\sin \alpha_m \cos \beta_m, \sin \alpha_m \sin \beta_m, \cos \alpha_m]^T$ . O sinal recebido no  $n_x$ -ésimo sensor que possui um vetor posição  $\mathbf{r}_{n_x} = [(n_x - 1)d_x, 0, 0]^T$  é dado por [16]

$$\begin{aligned} x_{n_x}(t) &= s_m(t)e^{-j\frac{2\pi}{\lambda}(\mathbf{k}_m^T \cdot \mathbf{r}_{n_x})} + v_{n_x}(t), \\ x_{n_x}(t) &= s_m(t)e^{-j\frac{2\pi}{\lambda}(n_x-1)d_x \sin \alpha_m \cos \beta_m} + v_{n_x}(t), \end{aligned} \quad (2.1)$$

em que  $v_{n_x}(t)$  é o ruído aditivo gaussiano branco proveniente do  $n_x$ -ésimo sensor. Para  $n_x = 1, \dots, N_x$  o vetor dos sinais recebidos no  $t$ -ésimo *snapshot* é descrito da seguinte forma

$$\mathbf{x}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_{N_x}(t) \end{bmatrix} = \begin{bmatrix} 1 \\ e^{-j\frac{2\pi}{\lambda}d_x \sin \alpha_m \cos \beta_m} \\ \vdots \\ e^{-j\frac{2\pi}{\lambda}(N_x-1)d_x \sin \alpha_m \cos \beta_m} \end{bmatrix} s_m(t) + \begin{bmatrix} v_1(t) \\ v_2(t) \\ \vdots \\ v_{N_x}(t) \end{bmatrix},$$

$$\mathbf{x}(t) = \mathbf{a}_x(\alpha_m, \beta_m)s_m(t) + \mathbf{v}_x(t) \in \mathbb{C}^{N_x \times 1}, \quad (2.2)$$

em que  $\mathbf{a}_x(\alpha_m, \beta_m) \in \mathbb{C}^{N_x \times 1}$  é o vetor de assinatura espacial da  $m$ -ésima fonte e  $\mathbf{v}_x(t) \in \mathbb{C}^{N_x \times 1}$  é o vetor que contém os ruídos oriundos dos sensores que compõem o arranjo linear uniforme posicionado no eixo- $x$ . Note que na equação (2.2) apenas uma fonte é considerada. Na prática o sinal recebido é

composto pela superposição de  $M$  sinais simultâneos e  $\mathbf{x}(t)$  pode ser reescrito como

$$\mathbf{x}(t) = \sum_{m=1}^M \mathbf{a}_x(\alpha_m, \beta_m) s_m(t) + \mathbf{v}_x(t) = \begin{bmatrix} \mathbf{a}_x(\alpha_1, \beta_1) & \cdots & \mathbf{a}_x(\alpha_M, \beta_M) \end{bmatrix} \begin{bmatrix} s_1(t) \\ \vdots \\ s_M(t) \end{bmatrix} + \mathbf{v}_x(t),$$

$$\mathbf{x}(t) = \mathbf{A}_x \mathbf{s}(t) + \mathbf{v}_x(t), \quad (2.3)$$

em que  $\mathbf{A}_x \in \mathbb{C}^{N_x \times M}$  é a matriz de assinaturas espaciais das  $M$  fontes obtida a partir do arranjo linear uniforme posicionado no eixo- $x$  e  $\mathbf{s}(t) \in \mathbb{C}^{M \times 1}$  é o vetor que contém as sequências transmitidas pelas fontes no  $t$ -ésimo *snapshot*.

Para completar a modelagem do receptor para um arranjo 2L-Shape é necessário descrever a recepção dos sinais nos demais eixos que compõem o arranjo. Dessa forma, o sinal da  $m$ -ésima fonte recebido no  $n_y$ -ésimo sensor que possui um vetor posição  $\mathbf{r}_{n_y} = [0, (n_y - 1) d_y, 0]^T$  é descrito pela seguinte expressão

$$y_{n_y}(t) = s_m(t) e^{-j \frac{2\pi}{\lambda} (\mathbf{k}_m^T \cdot \mathbf{r}_{n_y})} + v_{n_y}(t),$$

$$y_{n_y}(t) = s_m(t) e^{-j \frac{2\pi}{\lambda} (n_y - 1) d_y \sin \alpha_m \sin \beta_m} + v_{n_y}(t), \quad (2.4)$$

e para o  $n_z$ -ésimo sensor com vetor posição  $\mathbf{r}_{n_z} = [0, 0, (n_z - 1) d_z]^T$  temos

$$z_{n_z}(t) = s_m(t) e^{-j \frac{2\pi}{\lambda} (\mathbf{k}_m^T \cdot \mathbf{r}_{n_z})} + v_{n_z}(t),$$

$$z_{n_z}(t) = s_m(t) e^{-j \frac{2\pi}{\lambda} (n_z - 1) d_z \cos \alpha_m} + v_{n_z}(t), \quad (2.5)$$

em que  $v_{n_y}(t)$  e  $v_{n_x}(t)$  são os termos de ruído oriundos dos sensores considerados. Para  $n_y = 1, \dots, N_y$  e  $n_z = 1, \dots, N_z$  os vetores dos sinais recebidos nos eixos  $y$  e  $z$  no  $t$ -ésimo *snapshot* são dados respectivamente por

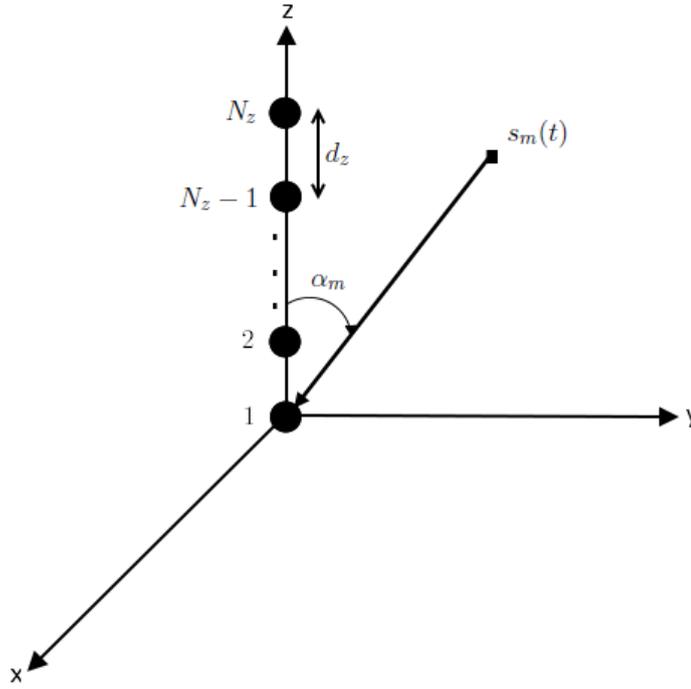
$$\mathbf{y}(t) = \begin{bmatrix} y_1(t) \\ y_2(t) \\ \vdots \\ y_{N_y}(t) \end{bmatrix} = \begin{bmatrix} 1 \\ e^{-j \frac{2\pi}{\lambda} d_y \sin \alpha_m \sin \beta_m} \\ \vdots \\ e^{-j \frac{2\pi}{\lambda} (N_y - 1) d_y \sin \alpha_m \sin \beta_m} \end{bmatrix} s_m(t) + \begin{bmatrix} v_1(t) \\ v_2(t) \\ \vdots \\ v_{N_y}(t) \end{bmatrix},$$

$$\mathbf{y}(t) = \mathbf{a}_y(\alpha_m, \beta_m) s_m(t) + \mathbf{v}_y(t) \in \mathbb{C}^{N_y \times 1}, \quad (2.6)$$

e

$$\mathbf{z}(t) = \begin{bmatrix} z_1(t) \\ z_2(t) \\ \vdots \\ z_{N_z}(t) \end{bmatrix} = \begin{bmatrix} 1 \\ e^{-j \frac{2\pi}{\lambda} d_z \cos \alpha_m} \\ \vdots \\ e^{-j \frac{2\pi}{\lambda} (N_z - 1) d_z \cos \alpha_m} \end{bmatrix} s_m(t) + \begin{bmatrix} v_1(t) \\ v_2(t) \\ \vdots \\ v_{N_z}(t) \end{bmatrix},$$

$$\mathbf{z}(t) = \mathbf{a}_z(\alpha_m) s_m(t) + \mathbf{v}_z(t) \in \mathbb{C}^{N_z \times 1}, \quad (2.7)$$



**Figura 2.2:** Arranjo Linear Uniforme com  $N_z$  sensores.

em que  $\mathbf{a}_y(\alpha_m, \beta_m) \in \mathbb{C}^{N_y \times 1}$ ,  $\mathbf{a}_z(\alpha_m) \in \mathbb{C}^{N_z \times 1}$ ,  $\mathbf{v}_y(t) \in \mathbb{C}^{N_y \times 1}$  e  $\mathbf{v}_z(t) \in \mathbb{C}^{N_z \times 1}$  são os vetores de assinaturas espaciais e de ruídos relacionados aos arranjos lineares posicionados nos eixos  $y$  e  $z$ , respectivamente. Da mesma forma que descrito anteriormente para  $M$  sinais simultâneos podemos reescrever  $\mathbf{y}(t)$  e  $\mathbf{z}(t)$  como

$$\mathbf{y}(t) = \sum_{m=1}^M \mathbf{a}_y(\alpha_m, \beta_m) s_m(t) + \mathbf{v}_y(t) \rightarrow \mathbf{y}(t) = \mathbf{A}_y \mathbf{s}(t) + \mathbf{v}_y(t), \quad (2.8)$$

e

$$\mathbf{z}(t) = \sum_{m=1}^M \mathbf{a}_z(\alpha_m) s_m(t) + \mathbf{v}_z(t) \rightarrow \mathbf{z}(t) = \mathbf{A}_z \mathbf{s}(t) + \mathbf{v}_z(t), \quad (2.9)$$

em que  $\mathbf{A}_y \in \mathbb{C}^{N_y \times M}$  e  $\mathbf{A}_z \in \mathbb{C}^{N_z \times M}$  são as matrizes de assinaturas espaciais referentes aos arranjos lineares posicionados nos eixos  $y$  e  $z$ , respectivamente. A partir da concatenação de (2.3), (2.8) e (2.9) obtemos os dados recebidos no arranjo 2L-Shape referentes ao  $t$ -ésimo *snapshot*, ou seja,

$$\underbrace{\begin{bmatrix} \mathbf{x}(t) \\ \mathbf{y}(t) \\ \mathbf{z}(t) \end{bmatrix}}_{\bar{\mathbf{x}}(t)} = \underbrace{\begin{bmatrix} \mathbf{A}_x \\ \mathbf{A}_y \\ \mathbf{A}_z \end{bmatrix}}_{\mathbf{A}} \mathbf{s}(t) + \underbrace{\begin{bmatrix} \mathbf{v}_x(t) \\ \mathbf{v}_y(t) \\ \mathbf{v}_z(t) \end{bmatrix}}_{\mathbf{v}(t)}, \quad (2.10)$$

$$\bar{\mathbf{x}}(t) = \mathbf{A} \mathbf{s}(t) + \mathbf{v}(t),$$

em que  $\bar{\mathbf{x}}(t) \in \mathbb{C}^{N \times 1}$ ,  $\mathbf{A} \in \mathbb{C}^{N \times M}$  e  $\mathbf{v}(t) \in \mathbb{C}^{N \times 1}$  e  $N = N_x + N_y + N_z$ .

**Caso Particular:** Um caso particular do arranjo *2L-Shape* bastante utilizado em estudos na área de processamento de sinais em arranjos é conhecido como **ULA** (do inglês, *Uniform Linear Array*) [2]. Esta configuração é obtida quando os sensores são posicionados uniformemente em apenas um dos eixos coordenados formando um arranjo 1D que proporciona realizar a estimação do ângulo de incidência dos sinais no arranjo em apenas uma direção. Por se tratar de uma versão simplificada do arranjo *2L-Shape*, o modelo de sinal para esta configuração é obtido utilizando apenas uma das etapas descritas anteriormente, por exemplo, para um arranjo linear uniforme posicionado no eixo- $z$  como ilustra a Figura 2.2 o modelo de sinal é descrito pela equação (2.9).

## 2.2 Modelo de Recepção de Sinais: Arranjo Planar

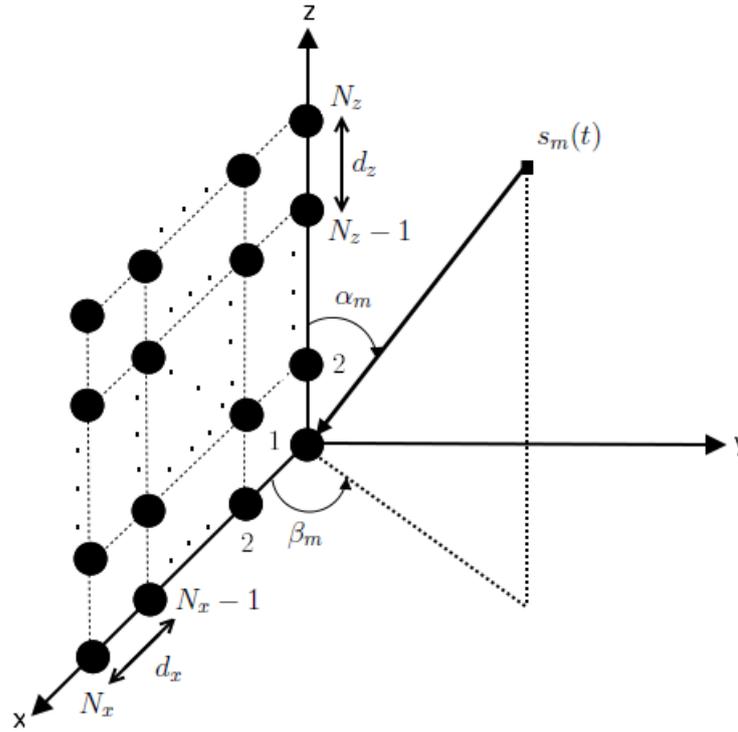
A segunda configuração abordada neste trabalho é denominada arranjo planar, formada por sensores que compõem as linhas e colunas do arranjo. Para descrever o modelo de sinal para esta configuração considere um arranjo posicionado no plano  $xz$  composto por  $N = N_x \cdot N_z$  sensores idênticos uniformemente espaçados de  $d_x$  em relação ao eixo- $x$  e  $d_z$  em relação ao eixo- $z$ , como ilustra a Figura 2.3. O sinal proveniente da  $m$ -ésima fonte recebido no  $(n_x, n_z)$ -ésimo sensor que possui um vetor posição  $\mathbf{r}_{n_x, n_z} = [(n_x - 1)d_x, 0, (n_z - 1)d_z]^T$  é dado por

$$\begin{aligned} x_{n_x, n_z}(t) &= s_m(t) e^{-j \frac{2\pi}{\lambda} (\mathbf{k}_m^T \cdot \mathbf{r}_{n_x, n_z})} + v_{n_x, n_z}(t), \\ x_{n_x, n_z}(t) &= s_m(t) e^{-j \frac{2\pi}{\lambda} [(n_x - 1)d_x \sin \alpha_m \cos \beta_m + (n_z - 1)d_z \cos \alpha_m]} + v_{n_x, n_z}(t), \end{aligned} \quad (2.11)$$

em que  $v_{n_x, n_z}(t)$  é o termo de ruído aditivo proveniente do  $(n_x, n_z)$ -ésimo sensor. O vetor dos sinais recebidos no  $t$ -ésimo *snapshot* proveniente de todos os sensores que compõem o arranjo planar, ou seja, para  $n_x = 1, \dots, N_x$  e  $n_z = 1, \dots, N_z$  é descrito da seguinte forma

$$\begin{aligned} \bar{\mathbf{x}}(t) = \begin{bmatrix} x_{1,1}(t) \\ x_{1,2}(t) \\ \vdots \\ x_{N_x, N_z}(t) \end{bmatrix} &= \begin{bmatrix} 1 \\ e^{-j \frac{2\pi}{\lambda} d_z \cos \alpha_m} \\ \vdots \\ e^{-j \frac{2\pi}{\lambda} [(N_x - 1)d_x \sin \alpha_m \cos \beta_m + (N_z - 1)d_z \cos \alpha_m]} \end{bmatrix} s_m(t) + \begin{bmatrix} v_{1,1}(t) \\ v_{1,2}(t) \\ \vdots \\ v_{N_x, N_z}(t) \end{bmatrix}, \\ \bar{\mathbf{x}}(t) &= \mathbf{a}_{xz}(\alpha_m, \beta_m) s_m(t) + \mathbf{v}(t) \in \mathbb{C}^{N \times 1}, \end{aligned} \quad (2.12)$$

em que  $\mathbf{a}_{xz}(\alpha_m, \beta_m) \in \mathbb{C}^{N \times 1}$  é o vetor de assinatura espacial do  $m$ -ésimo sinal recebido e  $\mathbf{v}(t) \in \mathbb{C}^{N \times 1}$  é o vetor de ruído aditivo, ambos relacionados ao arranjo planar. Assim como descrito no arranjo *2L-Shape*, o princípio da superposição



**Figura 2.3:** Arranjo Planar com  $N_x \cdot N_z$  sensores.

também é válido para o arranjo planar, dessa forma para  $M$  sinais incidentes simultaneamente podemos reescrever a equação (2.12) como

$$\bar{\mathbf{x}}(t) = \sum_{m=1}^M \mathbf{a}_{xz}(\alpha_m, \beta_m) s_m(t) + \mathbf{v}(t) \rightarrow \bar{\mathbf{x}}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{v}(t), \quad (2.13)$$

em que  $\mathbf{A} \in \mathbb{C}^{N \times M}$  é a matriz de assinaturas espaciais e seus elementos contém informação sobre a DoA dos sinais incidentes no arranjo de sensores e para a configuração planar temos que  $N = N_x \cdot N_z$ .

A partir das equações (2.10) e (2.13) é possível determinar os dados coletados nos arranjos *2L-Shape* e planar no  $t$ -ésimo *snapshot*. Porém, na prática é considerado um tempo de observação finito que possibilita a obtenção de  $T$  amostras do sinal recebido. Supondo que as matrizes de assinaturas espaciais não variam durante um intervalo de observação de  $t = 1, \dots, T$  *snapshots* então a matriz que contém os dados coletados durante esse intervalo é dada por

$$\mathbf{X} = \mathbf{A}\mathbf{S} + \mathbf{V}, \quad (2.14)$$

em que  $\mathbf{S} = [s(1), \dots, s(T)] \in \mathbb{C}^{M \times T}$  é a matriz que contém as sequências transmitidas pelas fontes,  $\mathbf{X} = [\bar{\mathbf{x}}(1), \dots, \bar{\mathbf{x}}(T)] \in \mathbb{C}^{N \times T}$  é a matriz que contém os dados coletados e  $\mathbf{V} = [\mathbf{v}(1), \dots, \mathbf{v}(T)] \in \mathbb{C}^{N \times T}$  é a matriz de ruídos e  $N = N_x + N_y + N_z$  para o arranjo *2L-Shape* e  $N = N_x \cdot N_z$  para o arranjo planar.

A equação (2.14) descreve um modelo matemático clássico em processamento de sinais em arranjos de sensores, em que diversos métodos matriciais foram desenvolvidos para realizar a estimação da DoA a partir dos dados coletados no arranjo. A seguir alguns desses métodos serão brevemente apresentados, entretanto um destaque maior será dado aos métodos MUSIC (do inglês, *Multiple Signal Classification*) e ESPRIT (do inglês, *Estimation of Signal Parameters by Rotational Invariance Techniques*) pois estes serão utilizados no Capítulo 4. Maiores detalhes sobre as demais técnicas assim como a descrição dos algoritmos podem ser encontrados em [2], [17] e [3].

## 2.3 Métodos Matriciais para Estimação de DoA

---

Nas últimas décadas diversas técnicas para estimação de DoA foram desenvolvidas. O sucesso de tais técnicas é proveniente da crescente utilização de arranjos de sensores em diversos campos como radar, sonar, astronomia e comunicações móveis. Em virtude desse vasto número de aplicações, essas técnicas vêm sendo amplamente estudadas e aperfeiçoadas com o objetivo de desenvolver algoritmos que apresentam uma alta resolução espacial e baixa complexidade computacional [18]. A seguir alguns desses métodos serão apresentados de acordo com suas divisões, ou seja, serão abordados os métodos baseados em estimação espacial, máxima verossimilhança e de forma mais detalhada os métodos baseados em subespaço MUSIC e ESPRIT.

Primeiramente, vamos abordar os métodos baseados em estimação espacial. Estes métodos são fundamentados numa busca exaustiva das direções de chegada dos sinais incidentes entre diversos valores possíveis e através do espectro espacial obtido é realizada a estimação da DoA que corresponde aos ângulos que geram os picos no espectro. Uma das técnicas mais simples para estimação de DoA é o método *Delay-and-Sum* [19]. Nesse método os sinais de saída das antenas são atrasados de forma que sua soma seja construtiva para uma certa direção [3]. Outro método baseado em estimação espacial é o Capon [20] e assim como no *Delay-and-Sum* ocorre uma busca exaustiva para possíveis valores de DoA. Entretanto, neste método os problemas de resolução associados ao *Delay-and-Sum* são superados pois a interferência proveniente de outras direções é minimizada [17]. Os métodos baseados em estimação espacial apresentam desempenho inferior aos métodos baseados em máxima verossimilhança e subespaço e maiores informações e detalhes sobre o desenvolvimento matemático do cálculo dos espectros para esses métodos podem ser encontrados em [17].

Os métodos baseados em máxima verossimilhança foram as primeiras técnicas a serem investigadas e sua principal desvantagem é seu elevado custo computacional. Entretanto, estes métodos apresentam desempenho superior em relação aos métodos baseados em subespaço principalmente em cenários que apresentam uma baixa SNR, nos casos em que o número de *snapshots* é reduzido e nos casos em que as fontes são correlacionadas [17]. Os métodos baseados em máxima verossimilhança são geralmente divididos em dois tipos, o determinístico e o estocástico, onde o segundo apresenta desempenho superior ao primeiro, entretanto, possui um custo computacional muito maior [18]. Detalhes sobre os métodos baseados em máxima verossimilhança e sumarização dos algoritmos para os casos determinístico e estocástico podem ser vistos em [17] e [3], respectivamente.

Por último, serão apresentados os métodos de estimação de DoA baseados em subespaço. Nestes métodos, as propriedades de autoestrutura da matriz de covariância dos dados coletados no arranjo de sensores são exploradas [3] de forma que seus autovetores são divididos em subespaço do sinal e subespaço do ruído, os quais são utilizados nos algoritmos. A estimação da DoA pode ser realizada de forma semelhante aos métodos baseados em estimação espacial utilizando o algoritmo MUSIC ou através de uma solução fechada utilizando o algoritmo ESPRIT. Dos diversos métodos de estimação de DoA existentes na literatura apenas os algoritmos MUSIC e ESPRIT foram utilizados nos resultados de simulação que serão apresentados no Capítulo 4, em virtude disso uma abordagem mais detalhada desses dois métodos será realizada a seguir.

### 2.3.1 Método MUSIC (*Multiple Signal Classification*)

Dos diversos métodos existentes para estimação de DoA sem dúvida um dos mais populares é o MUSIC, proposto por Schmidt [21] e verificado experimentalmente em [22]. Este método é baseado no cálculo da EVD (do inglês, *EigenValue Decomposition*) da matriz de covariância espacial dos dados recebidos no arranjo de sensores. Para descrevê-lo vamos assumir que os sinais transmitidos pelas diferentes fontes e o ruído aditivo proveniente dos sensores são descorrelacionados, de forma que a matriz de covariância espacial estimada dos dados coletados durante um intervalo de observação de  $t = 1, \dots, T$  *snapshots* seja descrita por

$$\mathbf{R} \triangleq \frac{1}{T} \mathbf{X} \mathbf{X}^H \rightarrow \mathbf{R} = \mathbf{A} \mathbf{R}_s \mathbf{A}^H + \sigma_v^2 \mathbf{I} \quad \text{quando } T \rightarrow \infty, \quad (2.15)$$

em que  $\mathbf{R}_s \triangleq \frac{1}{T} \mathbf{S} \mathbf{S}^H$  é a matriz de covariância das sequências transmitidas pelas fontes e  $\sigma_v^2$  é a variância do ruído. A partir da EVD da equação (2.15) podemos reescrever  $\mathbf{R}$  em termos dos subespaços do sinal e do ruído, ou seja,

$$\mathbf{R} = \mathbf{U}_s \mathbf{\Lambda}_s \mathbf{U}_s^H + \sigma_v^2 \mathbf{U}_v \mathbf{U}_v^H, \quad (2.16)$$

de forma que se  $\mathbf{A} \mathbf{R}_s \mathbf{A}^H$  possuir *rank* completo então  $\mathbf{\Lambda}_s$  é uma matriz diagonal formada pelos  $M$  maiores autovalores de  $\mathbf{R}$  e  $\mathbf{U}_s$  é a matriz denominada subespaço do sinal formada pelos autovetores correspondentes aos  $M$  maiores autovalores de  $\mathbf{R}$  e  $\mathbf{U}_v$  é a matriz denominada subespaço do ruído formada pelos autovetores correspondentes aos  $N - M$  menores autovalores de  $\mathbf{R}$ .

Se os autovetores que compõem a matriz  $\mathbf{U}_v$  forem ortogonais a matriz de assinaturas espaciais  $\mathbf{A}$ , isso implica que o vetor de assinatura espacial que possui a direção de chegada da  $m$ -ésima fonte encontra-se no subespaço do sinal. Dessa forma, realizando uma busca utilizando vetores formados pelas possíveis direções de chegada dos sinais incidentes no arranjo de sensores é possível determinar aqueles que são perpendiculares ao subespaço do ruído, sendo possível dessa forma, estimar a direção de chegada dos sinais recebidos.

Para um arranjo linear uniforme composto por  $N_z$  sensores como ilustra a Figura 2.2 em que o vetor de assinatura espacial  $\mathbf{a}(\alpha) \in \mathbb{C}^{N_z \times 1}$  descrito na equação (2.7) é função do ângulo de elevação, as direções de chegada dos sinais incidentes no arranjo podem ser estimadas localizando os valores de  $\alpha$  que correspondem aos  $M$  maiores picos do espectro espacial descrito por [2]:

$$P_{\text{MUSIC}}(\alpha) = \frac{\mathbf{a}^H(\alpha) \mathbf{a}(\alpha)}{\mathbf{a}^H(\alpha) \mathbf{U}_v \mathbf{U}_v^H \mathbf{a}(\alpha)}. \quad (2.17)$$

O denominador da equação (2.17) representa uma medida de distância entre o vetor de assinatura espacial  $\mathbf{a}(\alpha)$  e o subespaço do ruído  $\mathbf{U}_v$ , de forma que a ortogonalidade existente entre  $\mathbf{a}(\alpha)$  e  $\mathbf{U}_v$  minimiza essa distância, originando dessa forma, os  $M$  picos do espectro que correspondem às direções de chegada dos sinais incidentes no arranjo de sensores [17]. Para um arranjo linear uniforme, o algoritmo MUSIC será apresentado a seguir:

### Algoritmo MUSIC

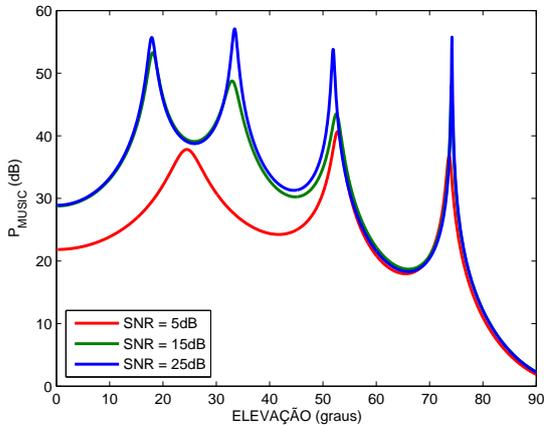
1. Crie um conjunto de valores iniciais para o ângulo de elevação  $\alpha$  para realizar a varredura do algoritmo;
2. A partir dos dados recebidos no decorrer dos  $t = 1, \dots, T$  *snapshots* considerados obtenha a matriz de covariância  $\mathbf{R} \triangleq \frac{1}{T} \mathbf{X} \mathbf{X}^H$ ;

3. A partir da EVD de  $\mathbf{R}$  obtenha o subespaço do ruído  $\mathbf{U}_v$  formado pelos autovetores correspondentes aos  $N - M$  menores autovalores de  $\mathbf{R}$ ;
4. Para cada valor de  $\alpha$  descrito no passo 1 calcule o espectro MUSIC:

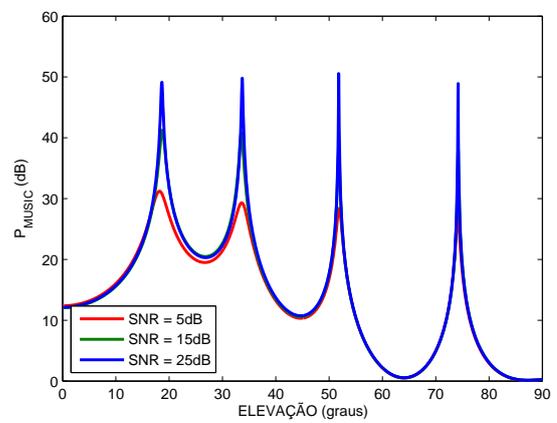
$$P_{\text{MUSIC}}(\alpha) = \frac{\mathbf{a}^H(\alpha)\mathbf{a}(\alpha)}{\mathbf{a}^H(\alpha)\mathbf{U}_v\mathbf{U}_v^H\mathbf{a}(\alpha)}$$

5. As estimações das direções de chegada são dadas pelos valores de  $\alpha$  que correspondem aos  $M$  maiores picos de  $P_{\text{MUSIC}}(\alpha)$ .

Exemplos de aplicações do algoritmo MUSIC para dois arranjos lineares uniformes distintos são ilustrados nas Figuras 2.4 e 2.5. Em ambos os casos as simulações foram realizadas utilizando os seguintes ângulos de elevação  $\alpha_1 = 18.6^\circ, \alpha_2 = 33.7^\circ, \alpha_3 = 51.8^\circ$  e  $\alpha_4 = 74.2^\circ$ . A partir dos resultados obtidos em cada caso é possível perceber que para  $N_z = 6$  a estimação do ângulo de elevação não foi realizada adequadamente para uma SNR igual a 5dB. Entretanto, quando o número de sensores aumenta para  $N_z = 10$  o desempenho obtido em todos os casos são semelhantes ocasionando praticamente uma superposição dos espectros espaciais para os valores de SNR considerados.



**Figura 2.4:** ULA com  $N_z = 6$  sensores.



**Figura 2.5:** ULA com  $N_z = 10$  sensores.

### 2.3.1.1 Algoritmo MUSIC para Arranjos 2L-Shape e Planar

Na literatura, diversos trabalhos utilizam o algoritmo MUSIC para estimar apenas os ângulos de elevação dos sinais incidentes, ou seja, na maioria dos trabalhos apenas arranjos lineares uniformes são considerados. Entretanto, este método pode ser facilmente estendido para situações em que se deseja estimar os ângulos de elevação e azimute simultaneamente [23] e [24]. Este procedimento é possível através da utilização de arranjos 2L-Shape e planares no receptor de forma que os vetores de assinatura espacial associados à DoA

descrita pelo par  $(\alpha, \beta)$  são dados por

$$\mathbf{a}(\alpha, \beta) = \begin{bmatrix} \mathbf{a}_x(\alpha, \beta) \\ \mathbf{a}_y(\alpha, \beta)_{(2:N_y)} \\ \mathbf{a}_z(\alpha)_{(2:N_z)} \end{bmatrix} \in \mathbb{C}^{N \times 1}, \quad (2.18)$$

para o arranjo *2L-Shape* e para o arranjo planar é descrito da seguinte forma

$$\mathbf{a}(\alpha, \beta) = \begin{bmatrix} 1 \\ e^{-j\frac{2\pi}{\lambda}d_z \cos \alpha_m} \\ \vdots \\ e^{-j\frac{2\pi}{\lambda}[(N_x-1)d_x \sin \alpha_m \cos \beta_m + (N_z-1)d_z \cos \alpha_m]} \end{bmatrix} \in \mathbb{C}^{N \times 1}, \quad (2.19)$$

em que  $N$  é o número de sensores que compõem os arranjos e pode ser diferente em cada caso. A partir dessas considerações o algoritmo MUSIC descrito anteriormente pode ser utilizado no processo de estimação, entretanto no caso 2D o passo 1 do algoritmo é levemente modificado de forma que um conjunto de valores para os ângulos  $\alpha$  e  $\beta$  são criados e utilizados no cálculo do espectro espacial MUSIC agora descrito por [23]:

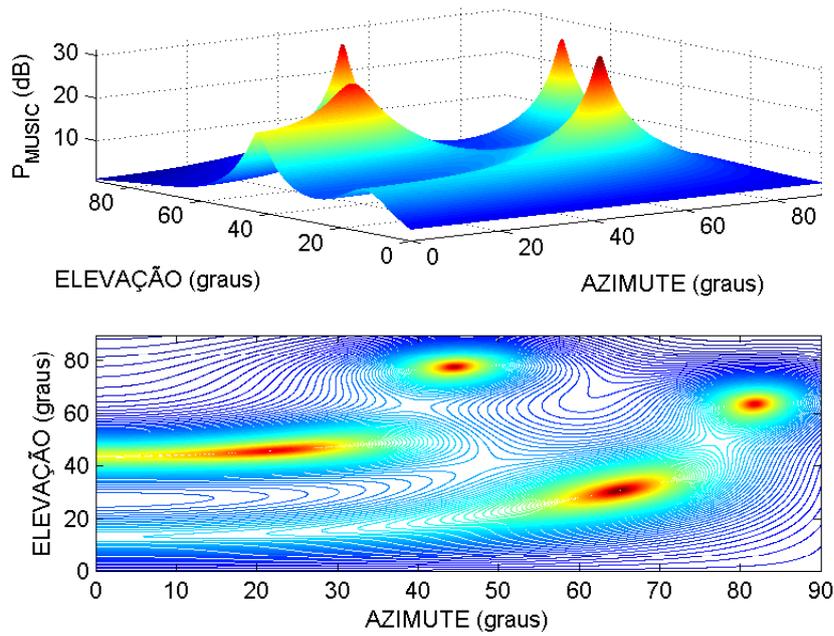
$$P_{\text{MUSIC}}(\alpha, \beta) = \frac{\mathbf{a}^H(\alpha, \beta)\mathbf{a}(\alpha, \beta)}{\mathbf{a}^H(\alpha, \beta)\mathbf{U}_v\mathbf{U}_v^H\mathbf{a}(\alpha, \beta)}, \quad (2.20)$$

de forma que as estimações das direções de chegada são obtidas pelos pares  $(\alpha, \beta)$  que correspondem aos  $M$  maiores picos de  $P_{\text{MUSIC}}(\alpha, \beta)$ .

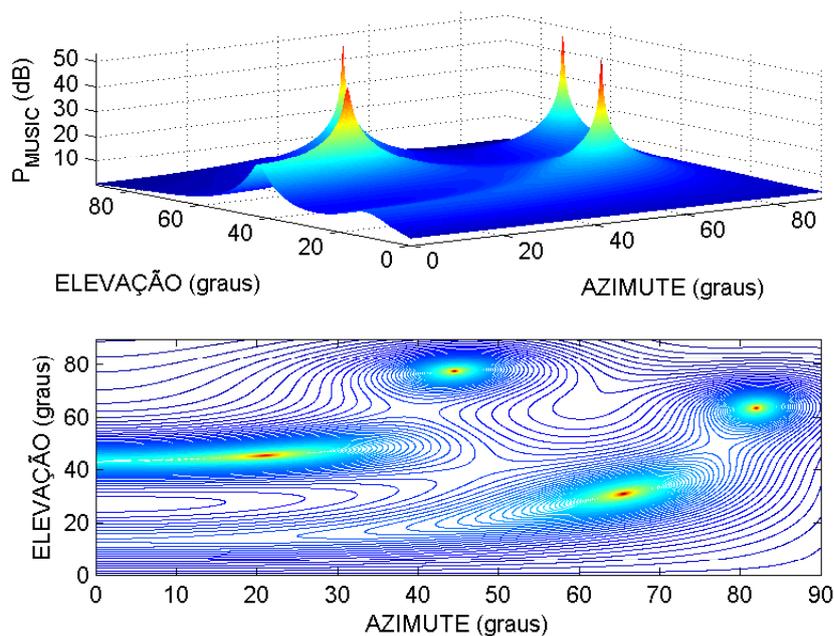
Assim como no caso do arranjo linear uniforme apresentado anteriormente, exemplos de aplicações do algoritmo MUSIC para arranjos planares são ilustrados nas Figuras 2.6 e 2.7. Nesses exemplos as simulações foram realizadas utilizando os seguintes pares  $(\alpha, \beta)$  para os ângulos de azimute e elevação dos sinais incidentes no arranjo  $(45.5^\circ, 21^\circ)$ ,  $(77.5^\circ, 44.5^\circ)$ ,  $(31^\circ, 65.5^\circ)$  e  $(63.5^\circ, 82^\circ)$  e o número de sensores considerados foi  $N_x = 4$  e  $N_z = 4$ . Como em ambos os casos foi considerado o mesmo número de sensores a diferença entre as simulações está na SNR utilizada de forma que a segunda simulação apresenta um desempenho superior a primeira em virtude do aumento da SNR de 10 dB para 25 dB sendo obtidos picos mais concentrados nesse caso.

### 2.3.1.2 Extensões do Algoritmo MUSIC

Em virtude do grande sucesso obtido pelo algoritmo MUSIC diversas pesquisas foram desenvolvidas com o intuito de melhorar sua resolução e diminuir sua complexidade computacional. Esse interesse proporcionou o desenvolvimento de algumas variações desse método e os algoritmos



**Figura 2.6:** Arranjo Planar com  $N_x = 4$ ,  $N_z = 4$  e SNR = 10dB.



**Figura 2.7:** Arranjo Planar com  $N_x = 4$ ,  $N_z = 4$  e SNR = 25dB.

*Cyclic-MUSIC* e *Root-MUSIC* foram propostos. O algoritmo *Cyclic-MUSIC* explora a propriedade de coerência espectral do sinal recebido assim como sua coerência espacial, proporcionando uma melhor solução para fontes com ângulos bem próximos. Além disso, esse método evita a necessidade de que o número de sinais incidentes seja menor que o número de sensores que compõem o arranjo [17].

A segunda extensão denominada *Root-MUSIC* foi proposta por Barabell [25] e apresenta melhor resolução que o MUSIC padrão principalmente para baixos valores de SNR [18]. Nesse algoritmo a análise do espectro com os possíveis valores de DoA é substituída pelo cálculo das raízes do polinômio  $D(z)$  equivalente ao denominador de (2.17) de modo que os picos do espectro são equivalentes as raízes de  $D(z)$  posicionadas próximo ao círculo unitário.

### 2.3.2 Método ESPRIT (*Estimation of Signal Parameters by Rotational Invariance Techniques*)

O processo de busca exaustiva existente nos métodos de estimação de DoA baseados em estimação espacial e no algoritmo MUSIC ocasionam um tempo computacional elevado dependendo da diferença entre os possíveis valores utilizados durante a varredura do espectro espacial. Em virtude disso, técnicas que reduzem o custo computacional foram desenvolvidas sendo uma das mais conhecidas o algoritmo ESPRIT desenvolvido por Roy et. al. [26] o qual é baseado na propriedade de invariância encontrada nos dados de saída dos sub-arranjos deslocados espacialmente. Diferentemente do MUSIC, o algoritmo ESPRIT possui uma solução fechada que proporciona uma redução considerável no custo computacional em comparação aos métodos espectrais em virtude da não realização de uma busca exaustiva entre os possíveis valores de azimuth e elevação dos sinais incidentes no arranjo.

Para descrever o algoritmo ESPRIT vamos considerar inicialmente que o arranjo linear uniforme posicionado no eixo- $z$  da Figura 2.1 seja dividido em dois sub-arranjos idênticos compostos por  $N_s = N_z - 1$  sensores cada. Nesse caso, a matriz de assinaturas espaciais relacionada ao segundo sub-arranjo é vista como uma versão defasada da matriz de assinaturas espaciais relacionada ao primeiro sub-arranjo, ou seja,

$$\mathbf{J}_1 \mathbf{A}_z \Phi_z = \mathbf{J}_2 \mathbf{A}_z, \quad (2.21)$$

nesse caso as matrizes  $\mathbf{J}_1 \in \mathbb{R}^{N_s \times N_z}$  e  $\mathbf{J}_2 \in \mathbb{R}^{N_s \times N_z}$  possuem elementos iguais a 0 ou 1 e são conhecidas como matrizes de seleção e são utilizadas para selecionar os sinais de saída dos sensores que compõem cada sub-arranjo definidas da seguinte forma

$$\mathbf{J}_1 = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 \end{bmatrix} \in \mathbb{R}^{N_s \times N_z} \quad \text{e} \quad \mathbf{J}_2 = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 1 \end{bmatrix} \in \mathbb{R}^{N_s \times N_z}, \quad (2.22)$$

e  $\Phi_z \in \mathbb{C}^{M \times M}$  é uma matriz diagonal que contém a defasagem de fase sofrida pelos sinais de saída do segundo sub-arranjo em relação aos sinais de saída do primeiro dada por

$$\Phi_z = \begin{bmatrix} e^{-j\frac{2\pi}{\lambda}d_z \cos \alpha_1} & & \\ & \ddots & \\ & & e^{-j\frac{2\pi}{\lambda}d_z \cos \alpha_M} \end{bmatrix}. \quad (2.23)$$

Assim como o MUSIC o algoritmo ESPRIT também é baseado em subespaço de forma que a matriz  $U_s \in \mathbb{C}^{N_z \times M}$  é obtida a partir do cálculo da EVD da matriz de covariância espacial  $R$  dos dados recebidos. Se o número de *snapshots* for suficientemente grande a matriz de assinaturas espaciais  $A_z$  está relacionada com o subespaço do sinal por  $A_z = U_s T$  em que  $T$  é uma matriz não-singular de dimensão  $M \times M$  de forma que a equação (2.21) pode ser reescrita como

$$J_1 U_s T \Phi_z = J_2 U_s T \rightarrow J_1 U_s \Psi_z = J_2 U_s, \quad (2.24)$$

em que  $\Psi_z = T \Phi_z T^{-1}$  é uma matriz não-singular com dimensão  $M \times M$ . Nesse caso se  $\Phi_z$  for uma matriz diagonal os elementos da diagonal principal são iguais aos autovalores de  $\Psi_z$ , sendo possível realizar a estimação do ângulo de elevação do  $m$ -ésimo sinal incidente calculando a EVD de  $\Psi_z$  e utilizando a seguinte relação

$$\hat{\alpha}_m = \arccos \left[ \frac{\arg(\Phi_{z(m,m)})}{2\pi d_z / \lambda} \right]. \quad (2.25)$$

O algoritmo ESPRIT também pode ser generalizado para o arranjo *2L-Shape* afim de se obter estimações conjuntas dos ângulos de azimuth e elevação. Para isso o mesmo procedimento realizado no eixo- $z$  pode ser repetido para os eixos  $x$  e  $y$  de forma semelhante a [14] obtendo dois sub-arranjos idênticos em cada eixo, compostos por  $N_x - 1$  e  $N_y - 1$  sensores respectivamente de forma que

$$J_1 A_x \Phi_x = J_2 A_x \quad \text{e} \quad J_1 A_y \Phi_y = J_2 A_y, \quad (2.26)$$

em que  $\Phi_x \in \mathbb{C}^{M \times M}$  e  $\Phi_y \in \mathbb{C}^{M \times M}$  são matrizes diagonais que contém a defasagem de fase sofrida pelos sinais de saída dos sub-arranjos deslocados nos eixos  $x$  e  $y$  em relação aos respectivos sub-arranjos de referência, descritas por

$$\Phi_x = \begin{bmatrix} e^{-j\frac{2\pi}{\lambda}d_x \sin \alpha_1 \cos \beta_1} & & \\ & \ddots & \\ & & e^{-j\frac{2\pi}{\lambda}d_x \sin \alpha_M \cos \beta_M} \end{bmatrix}, \quad (2.27)$$

e

$$\Phi_y = \begin{bmatrix} e^{-j\frac{2\pi}{\lambda}d_y \sin \alpha_1 \sin \beta_1} & & & \\ & \ddots & & \\ & & \ddots & \\ & & & e^{-j\frac{2\pi}{\lambda}d_y \sin \alpha_M \sin \beta_M} \end{bmatrix}, \quad (2.28)$$

e assim como descrito em (2.24) a equação (2.26) pode ser reescrita como

$$\mathbf{J}_{1x} \mathbf{U}_s \Psi_x = \mathbf{J}_{2x} \mathbf{U}_s \quad \text{e} \quad \mathbf{J}_{1y} \mathbf{U}_s \Psi_y = \mathbf{J}_{2y} \mathbf{U}_s, \quad (2.29)$$

em que  $\Psi_x$  e  $\Psi_y$  são matrizes não-singulares com autovalores iguais aos elementos da diagonal principal de  $\Phi_x$  e  $\Phi_y$ , respectivamente. Dessa forma, a partir de (2.29) é possível realizar a estimação do ângulo de azimute do sinal proveniente da  $m$ -ésima fonte a partir da EVD de  $\Psi_x$  e  $\Psi_y$  e utilizando as seguintes equações

$$\hat{\beta}_m^{(x)} = \arccos \left[ \frac{\arg(\Phi_{x(m,m)})}{(2\pi d_x \sin \hat{\alpha}_m) / \lambda} \right] \quad \text{e} \quad \hat{\beta}_m^{(y)} = \arcsin \left[ \frac{\arg(\Phi_{y(m,m)})}{(2\pi d_y \sin \hat{\alpha}_m) / \lambda} \right], \quad (2.30)$$

em que a estimação  $\hat{\alpha}_m$  do ângulo de elevação do  $m$ -ésimo sinal incidente no arranjo *2L-Shape* foi obtida em (2.25). Note que a utilização do arranjo *2L-Shape* no receptor proporciona realizar a estimação do ângulo de azimute duas vezes independentes obtendo  $\hat{\beta}_m^{(x)}$  e  $\hat{\beta}_m^{(y)}$ . Dessa forma, os resultados obtidos em cada caso podem ser combinados proporcionando uma diminuição no número de falhas durante o processo de estimação se o ângulo de azimute estimado for reescrito da seguinte forma de acordo com [15] e [27]:

$$\hat{\beta}_m = \begin{cases} \frac{1}{2} (\hat{\beta}_m^{(x)} + \hat{\beta}_m^{(y)}) & \text{se } \hat{\beta}_m^{(x)} \text{ e } \hat{\beta}_m^{(y)} \text{ forem reais} \\ \hat{\beta}_m^{(x)} & \text{se } \hat{\beta}_m^{(x)} \text{ for real e } \hat{\beta}_m^{(y)} \text{ for complexo} \\ \hat{\beta}_m^{(y)} & \text{se } \hat{\beta}_m^{(y)} \text{ for real e } \hat{\beta}_m^{(x)} \text{ for complexo} \\ \text{inválido} & \text{se ambos forem complexos.} \end{cases} \quad (2.31)$$

O algoritmo ESPRIT padrão para o arranjo *2L-Shape* será descrito de forma sumarizada a seguir:

### Algoritmo ESPRIT para o Arranjo *2L-Shape*

1. A partir dos dados recebidos nos arranjos lineares uniformes posicionados nos eixos  $x, y$  e  $z$  obtenha as matrizes de covariância  $\mathbf{R}_x$ ,  $\mathbf{R}_y$  e  $\mathbf{R}_z$ ;
2. Calcule independentemente a EVD de  $\mathbf{R}_x$ ,  $\mathbf{R}_y$  e  $\mathbf{R}_z$  e obtenha em cada caso o subespaço do sinal  $\mathbf{U}_s$ ;

## 3. Resolva

$$\mathbf{J}_{1x}\hat{\mathbf{U}}_s\Psi_x = \mathbf{J}_{2x}\mathbf{U}_s \quad , \quad \mathbf{J}_{1y}\mathbf{U}_s\Psi_y = \mathbf{J}_{2y}\mathbf{U}_s \quad \text{e} \quad \mathbf{J}_{1z}\mathbf{U}_s\Psi_z = \mathbf{J}_{2z}\mathbf{U}_s$$

por meio do método de mínimos quadrados;

4. Obtenha as matrizes diagonais  $\Phi_x$ ,  $\Phi_y$  e  $\Phi_z$  formadas pelos autovalores de  $\Psi_x$ ,  $\Psi_y$  e  $\Psi_z$ , respectivamente;
5. Obtenha para  $m = 1, \dots, M$  as estimações dos ângulos de elevação e azimute  $\hat{\alpha}_m$  e  $\hat{\beta}_m$  utilizando as equações (2.25), (2.30) e (2.31).

Note que o algoritmo ESPRIT descrito anteriormente também pode ser utilizado quando o receptor for formado por uma ULA como ilustra a Figura 2.2. Nesse caso apenas as operações relacionados ao arranjo linear uniforme posicionado no eixo- $z$  são realizadas, ou seja, nessa situação é necessário obter apenas os autovalores da matriz  $\Psi_z$  e realizar a estimação dos ângulos de elevação dos  $M$  sinais incidentes utilizando a equação (2.25).

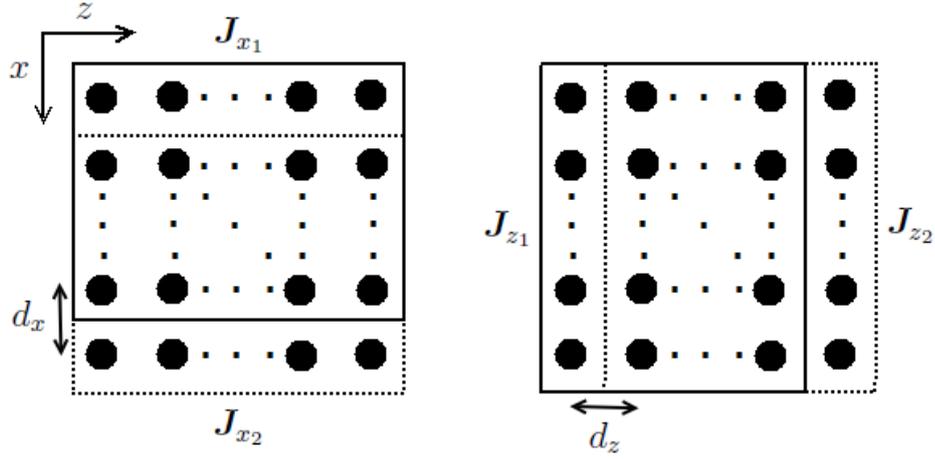
### 2.3.2.1 Algoritmo ESPRIT para o Arranjo Planar

Para descrever o algoritmo ESPRIT para um arranjo planar arbitrário vamos considerar que o arranjo ilustrado na Figura 2.3 seja dividido em dois sub-arranjos idênticos compostos por  $m_x$  sensores espaçados de  $d_x$  em relação ao eixo- $x$  e dois sub-arranjos idênticos compostos por  $m_z$  sensores espaçados de  $d_z$  em relação ao eixo- $z$ , obtendo ao todo quatro sub-arranjos planares, como ilustra a Figura 2.8. É possível perceber que de forma semelhante aos arranjos *2L-Shape* e ULA as matrizes de assinaturas espaciais relacionadas aos sub-arranjos deslocados podem ser interpretados como versões defasadas dos sub-arranjos de referência, ou seja,

$$\mathbf{J}_{x_1}\mathbf{A}\Phi_x = \mathbf{J}_{x_2}\mathbf{A} \quad \text{e} \quad \mathbf{J}_{z_1}\mathbf{A}\Phi_z = \mathbf{J}_{z_2}\mathbf{A}, \quad (2.32)$$

em que  $\mathbf{J}_{x_1}$ ,  $\mathbf{J}_{x_2}$ ,  $\mathbf{J}_{z_1}$  e  $\mathbf{J}_{z_2}$  são as matrizes de seleção utilizadas para selecionar os sensores que compõem cada sub-arranjo planar de acordo com a Figura 2.8 e podem ser facilmente obtidas a partir de  $\mathbf{J}_1$  e  $\mathbf{J}_2$  utilizando as seguintes relações [28]

$$\begin{aligned} \mathbf{J}_{x_1} &= \mathbf{I}_{N_z} \otimes \mathbf{J}_1^{(N_x)} & \text{e} & \quad \mathbf{J}_{x_2} = \mathbf{I}_{N_y} \otimes \mathbf{J}_2^{(N_x)}, \\ \mathbf{J}_{z_1} &= \mathbf{J}_1^{(N_z)} \otimes \mathbf{I}_{N_x} & \text{e} & \quad \mathbf{J}_{z_2} = \mathbf{J}_2^{(N_z)} \otimes \mathbf{I}_{N_x}. \end{aligned} \quad (2.33)$$



**Figura 2.8:** Divisão do Arranjo Planar em Sub-Arranjos.

De forma semelhante ao obtido para o arranjo *2L-Shape* apresentado de forma detalhada anteriormente a equação (2.32) pode ser reescrita como

$$J_{x_1} U_s \Psi_x = J_{x_2} U_s \quad \text{e} \quad J_{z_1} U_s \Psi_z = J_{z_2} U_s, \quad (2.34)$$

sendo possível estimar os ângulos de elevação e azimute das fontes seguindo os passos 3, 4 e 5 do algoritmo ESPRIT descrito na seção anterior. Nesse caso, o passo 3 segue uma estrutura semelhante a encontrada em [29] e [30].

### 2.3.2.2 Extensões do Algoritmo ESPRIT

Uma das extensões do algoritmo ESPRIT padrão é conhecida como *Unitary ESPRIT* e também válida para casos 1D e 2D. Este algoritmo utiliza matrizes unitárias esparsas e mantém a mesma estrutura do algoritmo ESPRIT padrão, porém possui um desempenho superior a este quando os sinais são correlacionados [31]. Outra vantagem do *Unitary ESPRIT* em comparação ao algoritmo padrão é a redução do custo computacional oriundo da realização de operações com valores reais principalmente nos casos em que as matrizes não apresentam dimensões elevadas.

Uma outra extensão do algoritmo ESPRIT conhecida como *Unitary ESPRIT Beamspace* é utilizada principalmente em aplicações que proporcionam informações a priori sobre as direções de chegada dos sinais incidentes, como por exemplo, em sistemas que utilizam radares [31]. Neste caso os cálculos podem ser realizados utilizando **DFT** (do inglês, *Discrete Fourier Transform*) nos setores de interesse de forma que a complexidade computacional do método é reduzida consideravelmente [28].

## 2.4 Fundamentos de Álgebra Multilinear e Decomposições Tensoriais

Nesta seção será realizada uma breve revisão sobre álgebra multilinear e decomposições tensoriais. Inicialmente, alguns conceitos fundamentais, formalismo matemático e principais notações e operações envolvendo tensores serão apresentadas. Em seguida, as decomposições tensoriais PARAFAC, Tucker e PARATUCK2 para tensores de ordem-3 e de ordens superiores assim como suas propriedades de unicidade serão abordadas. Por último, será realizado um levantamento dos diferentes trabalhos que propõem técnicas de estimação de DoA baseadas em álgebra multilinear e decomposições tensoriais. O conteúdo que será apresentado é de fundamental importância e será amplamente explorado no Capítulo 3 durante a descrição dos métodos tensoriais para estimação de DoA propostos neste trabalho.

### 2.4.1 Conceitos Básicos

O termo tensor vem sendo amplamente utilizado em diversas áreas do conhecimento possuindo significados distintos em cada uma, por exemplo, em Física, Matemática e Engenharia o termo está relacionado aos campos tensoriais e em quimiometria, psicometria e processamento de sinais o mesmo é utilizado como sinônimo de matrizes multidimensionais ou ordenamentos multidimensionais de dados [9], [32] e [33]. Por se tratar de uma ferramenta utilizada em diversos ramos muitos trabalhos abordando conceitos fundamentais referentes a álgebra multilinear foram desenvolvidos [9], [34] e [35] e algumas definições importantes para o desenvolvimento deste trabalho serão apresentadas a seguir.

**Definição 1.** Seja  $\mathcal{X} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$  um tensor de  $N$ -ésima ordem sua  $i_1, i_2, \dots, i_N$  entrada é dada por

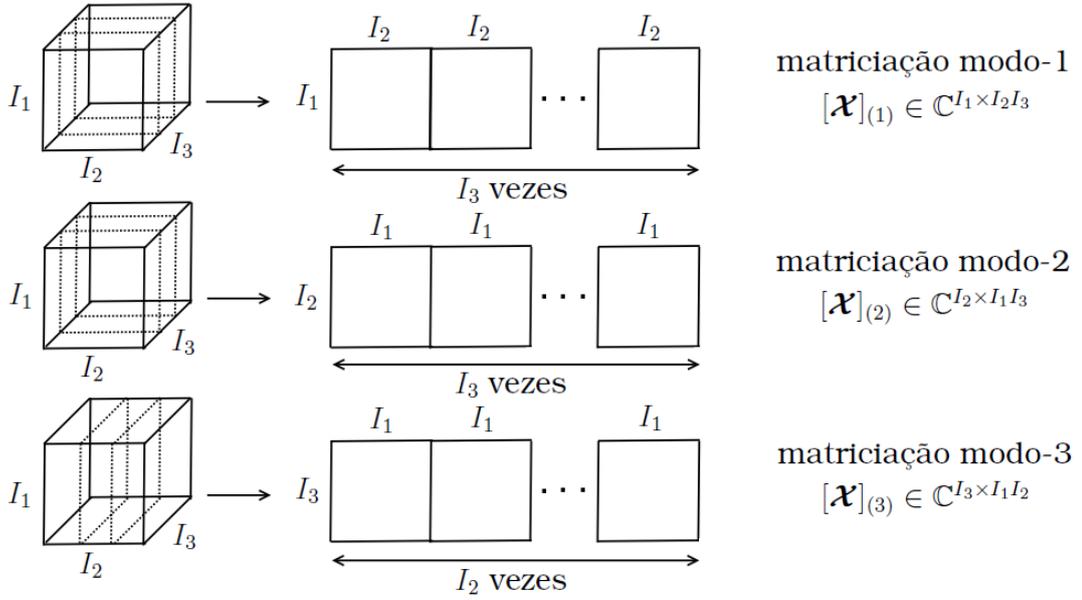
$$x_{i_1, i_2, \dots, i_N} = [\mathcal{X}]_{i_1, i_2, \dots, i_N}, \quad (2.35)$$

em que  $i_n$  é definido como sendo o  $n$ -ésimo modo ou dimensão do tensor  $\mathcal{X}$ .

**Definição 2.** O produto interno entre dois tensores  $\mathcal{X}$  e  $\mathcal{Y}$  de ordem  $N$  é definido da seguinte forma

$$\langle \mathcal{X}, \mathcal{Y} \rangle = \sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \dots \sum_{i_N=1}^{I_N} x_{i_1, i_2, \dots, i_N} y_{i_1, i_2, \dots, i_N}, \quad (2.36)$$

e quando  $\langle \mathcal{X}, \mathcal{Y} \rangle = 0$  os tensores são ditos ortogonais.



**Figura 2.9:** Matriciação de um Tensor de Terceira Ordem.

**Definição 3.** Um tensor  $\mathcal{X} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$  é denominado tensor de *rank-1* se o mesmo for obtido a partir do produto externo entre  $N$  vetores, ou seja,

$$\mathcal{X} = \mathbf{a}_1 \circ \mathbf{a}_2 \circ \dots \circ \mathbf{a}_N, \quad (2.37)$$

em que  $\mathbf{a}_n \in \mathbb{C}^{I_n}$  para  $n = 1, \dots, N$ . De forma geral o *rank* de um tensor arbitrário é definido como sendo o número mínimo de tensores de *rank-1* necessários para produzir  $\mathcal{X}$  a partir de combinações lineares.

**Definição 4.** Seja  $\mathcal{X} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$  um tensor de  $N$ -ésima ordem sua norma de Frobenius é dada por

$$\|\mathcal{X}\|_F = \sqrt{\langle \mathcal{X}, \mathcal{X} \rangle} = \sqrt{\sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \dots \sum_{i_N=1}^{I_N} |x_{i_1, i_2, \dots, i_N}|^2}. \quad (2.38)$$

A norma de Frobenius de um tensor é análoga a de uma matriz.

**Definição 5.** As operações matemáticas envolvendo decomposições tensoriais são realizadas de forma matricial, em virtude da obtenção de um melhor desempenho computacional [36]. Dessa forma, é necessário realizar uma reorganização dos dados multilineares de forma matricial sem que ocorra perda de informação. Esse procedimento é conhecido na literatura como processo de matriciação de um tensor e é baseado na concatenação dos seus *slices*, possibilitando a obtenção de diversas representações dos dados dependendo da ordem do tensor.

O processo de matriciação não é único [34] e no decorrer deste trabalho será utilizada a notação encontrada em [9] onde a forma matriciada do tensor  $\mathcal{X} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$  no modo- $n$  denotada por  $[\mathcal{X}]_{(n)} \in \mathbb{C}^{I_n \times I_1 I_2 \dots I_{n-1} I_{n+1} I_N}$  é obtida agrupando suas fibras modo- $n$  como colunas da matriz resultante. A Figura 2.9 ilustra o processo de matriciação do tensor  $\mathcal{X} \in \mathbb{C}^{I_1 \times I_2 \times I_3}$ .

**Definição 6.** O produto modo- $n$  de um tensor  $\mathcal{X} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$  com uma matriz  $A \in \mathbb{C}^{J \times I_n}$  em relação ao  $n$ -ésimo modo de  $\mathcal{X}$  é denotado por  $\mathcal{Y} = \mathcal{X} \times_n A$  e o resultado da operação é o tensor  $\mathcal{Y} \in \mathbb{C}^{I_1 \times \dots \times I_{n-1} \times J \times I_{n+1} \times \dots \times I_N}$  em que seus elementos são dados por

$$(\mathcal{X} \times_n A)_{i_1 \dots i_{n-1} j i_{n+1} \dots i_N} = \sum_{i_n=1}^{I_n} x_{i_1 i_2 \dots i_N} a_{j i_n}. \quad (2.39)$$

Nesse produto cada fibra modo- $n$  de  $\mathcal{X}$  é multiplicada pela matriz  $A$ , podendo ser descrito de forma equivalente como  $[\mathcal{Y}]_{(n)} = A[\mathcal{X}]_{(n)}$  representando uma mudança de base no tensor  $\mathcal{X}$  [37].

**Definição 7.** O produto de Kronecker entre as matrizes  $A \in \mathbb{C}^{I \times J}$  e  $B \in \mathbb{C}^{K \times L}$  é denotado por  $A \otimes B$  e o resultado é uma matriz de tamanho  $IK \times JL$  definida como

$$A \otimes B = \begin{bmatrix} a_{11}B & \dots & a_{1J}B \\ \vdots & \ddots & \vdots \\ a_{I1}B & \dots & a_{IJ}B \end{bmatrix} \in \mathbb{C}^{IK \times JL}. \quad (2.40)$$

**Definição 8.** O produto de Khatri-Rao entre as matrizes  $A \in \mathbb{C}^{I \times K}$  e  $B \in \mathbb{C}^{J \times K}$  é denotado por  $A \diamond B$  e o resultado é uma matriz de tamanho  $IJ \times K$  definida como

$$A \diamond B = \begin{bmatrix} a_1 \otimes b_1 & a_2 \otimes b_2 & \dots & a_K \otimes b_K \end{bmatrix} \in \mathbb{C}^{IJ \times K}, \quad (2.41)$$

em que  $a_k$  e  $b_k$  para  $k = 1, \dots, K$  são as colunas das matrizes  $A$  e  $B$ . Os produtos de Kronecker e Khatri-Rao são utilizados para descrever as formas matriciadas dos tensores.

### 2.4.2 Decomposições Tensoriais

As decomposições tensoriais vêm sendo amplamente utilizadas em diversas áreas, principalmente no que diz respeito a solução de problemas que envolvem misturas multilineares, proporcionando identificar e extrair os diferentes fatores existentes a partir de um conjunto de dados medidos. A ideia inicial a respeito das decomposições tensoriais é datada de 1927, originada

**Figura 2.10:** Decomposição PARAFAC de um Tensor de Terceira Ordem.

por Hitchcock [38] e posteriormente desenvolvida por Cattell em 1944 [39] e por Tucker em 1966 [40] ambas no contexto da Psicometria, e atualmente vêm sendo utilizadas com sucesso em processamento de sinais em virtude de suas propriedades de unicidade total ou parcial. A seguir as decomposições tensoriais PARAFAC, Tucker e PARATUCK2 serão apresentadas, estas serão utilizadas no decorrer deste trabalho principalmente no Capítulo 3 onde os algoritmos propostos para estimação de DoA serão descritos.

### 2.4.2.1 Decomposição PARAFAC

A decomposição **PARAFAC** (do inglês, *Parallel Factor Analysis*) também conhecida como **CANDECOMP** (do inglês, *Canonical Decomposition*) de um tensor de terceira ordem foi proposta independentemente em 1970 por Harshman [41] e Carroll e Chang [42] no contexto da Psicometria. Para um tensor de terceira ordem  $\mathcal{X} \in \mathbb{C}^{I_1 \times I_2 \times I_3}$  a decomposição PARAFAC decompõe o  $(i_1, i_2, i_3)$ -ésimo elemento de  $\mathcal{X}$  em componentes trilineares de forma que a representação escalar dessa decomposição é dada por

$$x_{i_1, i_2, i_3} = \sum_{r=1}^R a_{i_1, r} b_{i_2, r} c_{i_3, r}, \quad (2.42)$$

em que  $x_{i_1, i_2, i_3} = [\mathcal{X}]_{i_1, i_2, i_3}$  é o  $(i_1, i_2, i_3)$ -ésimo elemento do tensor  $\mathcal{X}$  e  $a_{i_1, r} = [\mathbf{A}]_{i_1, r}$ ,  $b_{i_2, r} = [\mathbf{B}]_{i_2, r}$  e  $c_{i_3, r} = [\mathbf{C}]_{i_3, r}$  são os elementos das matrizes fatores  $\mathbf{A} \in \mathbb{C}^{I_1 \times R}$ ,  $\mathbf{B} \in \mathbb{C}^{I_2 \times R}$  e  $\mathbf{C} \in \mathbb{C}^{I_3 \times R}$ , respectivamente e  $R$  é um inteiro positivo definido como sendo o *rank* do tensor  $\mathcal{X}$  [9]. Outra forma bastante comum de representar a decomposição PARAFAC é obtida em termos do produto externo onde um tensor de *rank*  $R$  é descrito como uma soma de  $R$  tensores de *rank-1*, como ilustra a Figura 2.10

$$\mathcal{X} = \sum_{r=1}^R \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r, \quad (2.43)$$

em que  $\mathbf{a}_r$ ,  $\mathbf{b}_r$  e  $\mathbf{c}_r$  são as  $r$ -ésimas colunas das matrizes fatores  $\mathbf{A}$ ,  $\mathbf{B}$  e  $\mathbf{C}$ , respectivamente.

Como dito anteriormente os dados que compõem um tensor podem ser organizados matricialmente sem que ocorra perda de informação utilizando o processo de matriciação descrito na **Definição 5**. Para um tensor de terceira ordem existem três formas distintas de realizar seu fatiamento, ou seja, fixando duas dimensões e variando uma, obtendo dessa forma a seguinte notação para os *slices* do tensor  $\mathcal{X}$  conforme [33] e [36]

$$\begin{aligned} \mathbf{X}_{i_1..} &= \mathbf{B}D_{i_1}(\mathbf{A})\mathbf{C}^T, & i_1 &= 1, \dots, I_1. \\ \mathbf{X}_{..i_2} &= \mathbf{C}D_{i_2}(\mathbf{B})\mathbf{A}^T, & i_2 &= 1, \dots, I_2. \\ \mathbf{X}_{..i_3} &= \mathbf{A}D_{i_3}(\mathbf{C})\mathbf{B}^T, & i_3 &= 1, \dots, I_3. \end{aligned} \quad (2.44)$$

em que  $D_{i_1}(\mathbf{A})$  é uma matriz diagonal formada pela  $i_1$ -ésima linha de  $\mathbf{A}$ . Realizando a concatenação coluna a coluna das estruturas obtidas anteriormente é possível determinar as formas matriciadas do tensor  $\mathcal{X}$  obtendo as seguintes representações

$$\begin{aligned} [\mathcal{X}]_{(1)} &= \begin{bmatrix} \mathbf{X}_{..1} & \cdots & \mathbf{X}_{..I_3} \end{bmatrix} = \mathbf{A} \begin{bmatrix} D_1(\mathbf{C})\mathbf{B}^T & \cdots & D_{I_3}(\mathbf{C})\mathbf{B}^T \end{bmatrix} \\ [\mathcal{X}]_{(1)} &= \mathbf{A}(\mathbf{C} \diamond \mathbf{B})^T \in \mathbb{C}^{I_1 \times I_2 I_3}. \end{aligned} \quad (2.45)$$

$$\begin{aligned} [\mathcal{X}]_{(2)} &= \begin{bmatrix} \mathbf{X}_{..1}^T & \cdots & \mathbf{X}_{..I_3}^T \end{bmatrix} = \mathbf{B} \begin{bmatrix} D_1(\mathbf{C})\mathbf{A}^T & \cdots & D_{I_3}(\mathbf{C})\mathbf{A}^T \end{bmatrix} \\ [\mathcal{X}]_{(2)} &= \mathbf{B}(\mathbf{C} \diamond \mathbf{A})^T \in \mathbb{C}^{I_2 \times I_1 I_3}. \end{aligned} \quad (2.46)$$

$$\begin{aligned} [\mathcal{X}]_{(3)} &= \begin{bmatrix} \mathbf{X}_{.1.} & \cdots & \mathbf{X}_{.I_2.} \end{bmatrix} = \mathbf{C} \begin{bmatrix} D_1(\mathbf{B})\mathbf{A}^T & \cdots & D_{I_2}(\mathbf{B})\mathbf{A}^T \end{bmatrix} \\ [\mathcal{X}]_{(3)} &= \mathbf{C}(\mathbf{B} \diamond \mathbf{A})^T \in \mathbb{C}^{I_3 \times I_1 I_2}. \end{aligned} \quad (2.47)$$

As equações (2.45), (2.46) e (2.47) relacionam as diversas formas de organização do tensor de dados obtidas a partir do processo de matriciação de  $\mathcal{X}$  com as matrizes fatores  $\mathbf{A}$ ,  $\mathbf{B}$  e  $\mathbf{C}$ . Essas relações serão utilizadas nos algoritmos que serão descritos detalhadamente no Capítulo 3.

Um dos principais motivos da popularidade da decomposição PARAFAC está relacionado a sua propriedade de unicidade que torna possível a obtenção de uma decomposição única para tensores de ordem três ou superior. Por esse motivo, diversos trabalhos abordando esse assunto foram publicados, por exemplo, [41] e [43] e uma condição suficiente para alcançar a unicidade na decomposição de tensores de ordem-3 com valores reais foi obtida por Kruskal [44], generalizada para tensores com valores complexos em [45] e posteriormente para tensores de ordem  $N$  em [46]. A condição de unicidade da decomposição PARAFAC está relacionada com o  $k$ -rank (*Kruskal-rank*) das matrizes fatores sendo este definido como:

**Definição 9.** Seja  $A \in \mathbb{C}^{I_1 \times R}$  o  $k$ -rank de  $A$  denotado por  $k_A$  é um número máximo  $k$  para o qual todo conjunto de  $k$  colunas de  $A$  é linearmente independente. A partir da definição do  $k$ -rank das matrizes fatores, a propriedade de unicidade de um tensor de terceira ordem é satisfeita quando

$$k_A + k_B + k_C \geq 2R + 2, \quad (2.48)$$

e nesse caso as matrizes  $A$ ,  $B$  e  $C$  são únicas mesmo existindo ambiguidade de permutação entre colunas e fator de escala, ou seja, quaisquer matrizes  $\tilde{A}$ ,  $\tilde{B}$  e  $\tilde{C}$  estão relacionadas com  $A$ ,  $B$  e  $C$  por:

$$\tilde{A} = A\Pi\Delta_1, \quad \tilde{B} = B\Pi\Delta_2, \quad \tilde{C} = C\Pi\Delta_3, \quad (2.49)$$

em que  $\Pi$  é uma matriz de permutação e  $\Delta_1$ ,  $\Delta_2$  e  $\Delta_3$  são matrizes diagonais que satisfazem a seguinte relação

$$\Delta_1\Delta_2\Delta_3 = I_R. \quad (2.50)$$

Os casos mais comuns encontrados na literatura abordam decomposições de tensores de terceira ordem, entretanto a decomposição PARAFAC assim como outras decomposições tensoriais podem ser generalizadas para tensores de ordem  $N$ . Seja  $\mathcal{X} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$  sua decomposição PARAFAC decompõem o  $(i_1, i_2, \dots, i_N)$ -ésimo elemento de  $\mathcal{X}$  em componentes multilineares dadas por

$$x_{i_1, i_2, \dots, i_N} = \sum_{r=1}^R a_{i_1, r}^{(1)} a_{i_2, r}^{(2)} \dots a_{i_N, r}^{(N)} \quad (2.51)$$

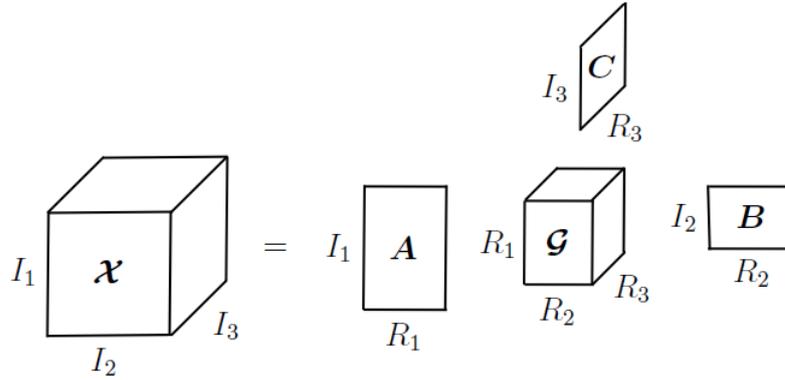
em que  $a_{i_n, r}^{(n)} = [A^{(n)}]_{i_n, r}$  para  $i_n = 1, \dots, I_n$  e  $n = 1, \dots, N$  são os elementos da  $n$ -ésima matriz fator da decomposição. De forma generalizada a matriciação modo- $n$  do tensor  $\mathcal{X}$  pode ser obtida utilizando a seguinte relação [9]

$$[\mathcal{X}]_{(n)} = A^{(n)} (A^{(N)} \diamond \dots \diamond A^{(n+1)} \diamond A^{(n-1)} \diamond \dots \diamond A^{(1)})^T, \quad n = 1, \dots, N. \quad (2.52)$$

Como dito anteriormente a propriedade de unicidade da decomposição PARAFAC também é válida para tensores de ordem  $N$  de acordo com a seguinte condição suficiente proposta em [46]

$$\sum_{n=1}^N k_{A^{(n)}} \geq 2R + (N - 1), \quad (2.53)$$

note que a condição anterior é igual a equação (2.48) quando  $N = 3$ .



**Figura 2.11:** Decomposição Tucker3.

### 2.4.2.2 Decomposição Tucker

A decomposição Tucker foi proposta por Ledyard Tucker em 1966 [40] e serviu como base para a elaboração de outros modelos que incorporam algumas de suas características, tornando-a a decomposição tensorial que possui o maior número de casos especiais. Para um tensor de terceira ordem  $\mathcal{X} \in \mathbb{C}^{I_1 \times I_2 \times I_3}$  a decomposição Tucker decompõem o  $(i_1, i_2, i_3)$ -ésimo elemento de  $\mathcal{X}$  em componentes trilineares ponderadas pelo elemento  $g_{r_1, r_2, r_3}$  que determina o nível de interação entre as diferentes componentes [9] e sua representação escalar é descrita como

$$x_{i_1, i_2, i_3} = \sum_{r_1=1}^{R_1} \sum_{r_2=1}^{R_2} \sum_{r_3=1}^{R_3} g_{r_1, r_2, r_3} a_{i_1, r_1} b_{i_2, r_2} c_{i_3, r_3}, \quad (2.54)$$

em que  $a_{i_1, r_1}$ ,  $b_{i_2, r_2}$  e  $c_{i_3, r_3}$  são os elementos das matrizes fatores  $A \in \mathbb{C}^{I_1 \times R_1}$ ,  $B \in \mathbb{C}^{I_2 \times R_2}$  e  $C \in \mathbb{C}^{I_3 \times R_3}$  e  $g_{r_1, r_2, r_3}$  é o  $(r_1, r_2, r_3)$ -ésimo elemento do tensor núcleo  $\mathcal{G} \in \mathbb{C}^{R_1 \times R_2 \times R_3}$ . A decomposição Tucker também pode ser reescrita utilizando a notação de produto modo- $n$  definida em (2.39) da seguinte forma

$$\mathcal{X} = \mathcal{G} \times_1 A \times_2 B \times_3 C, \quad (2.55)$$

e as formas matriciadas do tensor  $\mathcal{X}$  referentes aos modos 1, 2 e 3 são dadas respectivamente por

$$\begin{aligned} [\mathcal{X}]_{(1)} &= A[\mathcal{G}]_{(1)} (C \otimes B)^T \in \mathbb{C}^{I_1 \times I_2 I_3}, \\ [\mathcal{X}]_{(2)} &= B[\mathcal{G}]_{(2)} (C \otimes A)^T \in \mathbb{C}^{I_2 \times I_1 I_3}, \\ [\mathcal{X}]_{(3)} &= C[\mathcal{G}]_{(3)} (B \otimes A)^T \in \mathbb{C}^{I_3 \times I_1 I_2}. \end{aligned} \quad (2.56)$$

O modelo descrito em (2.54) e (2.55) e ilustrado na Figura 2.11 é conhecido como decomposição Tucker3 e representa uma das diversas variações da

decomposição Tucker, definida quando nenhuma das matrizes fatores é identidade. Para um tensor de terceira ordem as demais representações da decomposição Tucker são obtidas quando uma ou duas matrizes fatores são iguais a identidade. Para o caso em que a matriz fator  $C = I_{I_3}$ , temos a descrição da decomposição Tucker2 dada por

$$\mathcal{X} = \mathcal{G} \times_1 A \times_2 B, \quad (2.57)$$

em que  $R_3 = I_3$  e  $\mathcal{G} \in \mathbb{C}^{R_1 \times R_2 \times I_3}$ . Assim como a decomposição Tucker2 a decomposição Tucker1 também é originada da decomposição Tucker3 e ocorre quando  $B = I_{I_2}$  e  $C = I_{I_3}$ , sendo descrita em notação de produto modo- $n$  da seguinte forma

$$\mathcal{X} = \mathcal{G} \times_1 A, \quad (2.58)$$

em que  $R_2 = I_2$ ,  $R_3 = I_3$  e  $\mathcal{G} \in \mathbb{C}^{R_1 \times I_2 \times I_3}$ .

Diferentemente da decomposição PARAFAC a decomposição Tucker3 permite a interação entre as componentes nos três modos do tensor  $\mathcal{X}$ , fato que a torna menos restrita que a decomposição PARAFAC. Porém, ambas estão relacionadas, e conceitualmente a decomposição PARAFAC de um tensor de terceira ordem pode ser vista como um caso especial da decomposição Tucker3 quando  $R_1 = R_2 = R_3 = R$  e o tensor núcleo  $\mathcal{G}$  for superdiagonal [9]. Nesse caso a propriedade de unicidade é válida de acordo com a equação (2.48) e a decomposição PARAFAC pode ser definida em notação de produto modo- $n$  como

$$\mathcal{X} = \mathcal{I}_{3,R} \times_1 A \times_2 B \times_3 C. \quad (2.59)$$

No decorrer deste trabalho serão utilizadas decomposições Tucker de ordem superior e assim como descrito para a decomposição PARAFAC é possível obter sua generalização para tensores de ordem  $N$ . Considerando  $\mathcal{X} \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$  a forma escalar da decomposição Tucker de  $N$ -ésima ordem do tensor  $\mathcal{X}$  é dada por [47]

$$x_{i_1, i_2, \dots, i_N} = \sum_{r_1=1}^{R_1} \sum_{r_2=1}^{R_2} \dots \sum_{r_N=1}^{R_N} g_{r_1, r_2, \dots, r_N} a_{i_1, r_1}^{(1)} a_{i_2, r_2}^{(2)} \dots a_{i_N, r_N}^{(N)}, \quad (2.60)$$

em que  $a_{i_n, r_n}^{(n)}$  é um elemento da  $n$ -ésima matriz fator da decomposição e  $g_{r_1, r_2, \dots, r_N}$  é o  $(r_1, r_2, \dots, r_N)$ -ésimo elemento do tensor núcleo  $\mathcal{G} \in \mathbb{C}^{R_1 \times R_2 \times \dots \times R_N}$  para  $i_n = 1, \dots, I_n$  e  $n = 1, \dots, N$  respectivamente. A equação (2.60) pode ser reescrita utilizando produto modo- $n$  da seguinte forma

$$\mathcal{X} = \mathcal{G} \times_1 A^{(1)} \times_2 A^{(2)} \dots \times_N A^{(N)}, \quad n = 1, \dots, N. \quad (2.61)$$

$$\begin{array}{c}
 I_1 \boxed{X_{\dots i_3}} \\
 I_2
 \end{array}
 =
 \begin{array}{c}
 I_1 \boxed{A} \\
 R_1
 \end{array}
 \begin{array}{c}
 D_{i_3}(C^A) \\
 \boxed{\text{---}} \\
 \text{---}
 \end{array}
 \begin{array}{c}
 R_1 \boxed{G} \\
 R_2
 \end{array}
 \begin{array}{c}
 D_{i_3}(C^B) \\
 \boxed{\text{---}} \\
 \text{---}
 \end{array}
 \begin{array}{c}
 R_2 \boxed{B^T} \\
 I_2
 \end{array}$$

**Figura 2.12:** Decomposição PARATUCK2.

De forma generalizada, as formas matriciadas de  $\mathcal{X}$  podem ser obtidas utilizando a seguinte relação [9]

$$[\mathcal{X}]_{(n)} = \mathbf{A}^{(n)} [\mathcal{G}]_{(n)} (\mathbf{A}^{(N)} \otimes \dots \otimes \mathbf{A}^{(n+1)} \otimes \mathbf{A}^{(n-1)} \otimes \dots \otimes \mathbf{A}^{(1)})^T. \quad (2.62)$$

Ao contrário da decomposição PARAFAC a decomposição Tucker não possui unicidade, ou seja, existem infinitas soluções para as matrizes fatores e para o tensor núcleo que recaem numa reconstrução perfeita do tensor de dados  $\mathcal{X}$ , uma prova pode ser vista em [33]. A unicidade completa da decomposição Tucker é possível em alguns casos especiais e a unicidade parcial da decomposição Tucker3 é discutida com detalhes em [48].

### 2.4.2.3 Decomposição PARATUCK2

Diversas decomposições tensoriais foram desenvolvidas a partir dos modelos clássicos PARAFAC e Tucker. Tais decomposições são baseadas principalmente na existência de simetria entre dois modos de um tensor [42], irregularidade no conjunto de dados e variação simultânea de duas matrizes fatores [49] e restrições lineares nas matrizes fatores [50]. Entretanto as características das decomposições PARAFAC e Tucker podem ser combinadas gerando uma decomposição híbrida. A mais comum é conhecida como decomposição PARATUCK2, proposta por Harshman e Lundy em [51], adicionando numa mesma decomposição a flexibilidade do Tucker2 e mantendo algumas propriedades de unicidade da decomposição PARAFAC. Para um tensor de terceira ordem  $\mathcal{X} \in \mathbb{C}^{I_1 \times I_2 \times I_3}$  sua decomposição PARATUCK2 é descrita em notação escalar da seguinte forma

$$x_{i_1, i_2, i_3} = \sum_{r_1=1}^{R_1} \sum_{r_2=1}^{R_2} a_{i_1, r_1} b_{i_2, r_2} g_{r_1, r_2} c_{i_3, r_1}^A c_{i_3, r_2}^B, \quad (2.63)$$

em que  $a_{i_1, r_1}$  e  $b_{i_2, r_2}$  são os elementos das matrizes fatores  $\mathbf{A} \in \mathbb{C}^{I_1 \times R_1}$  e  $\mathbf{B} \in \mathbb{C}^{I_2 \times R_2}$  associadas aos modos 1 e 2 de  $\mathcal{X}$ ,  $c_{i_3, r_1}^A$  e  $c_{i_3, r_2}^B$  são os elementos das matrizes de interação  $\mathbf{C}^A \in \mathbb{C}^{I_3 \times R_1}$  e  $\mathbf{C}^B \in \mathbb{C}^{I_3 \times R_2}$  que definem uma combinação linear entre as colunas de  $\mathbf{A}$  e  $\mathbf{B}$  ao longo do terceiro modo de  $\mathcal{X}$  e  $g_{r_1, r_2}$  são

os elementos de  $\mathbf{G} \in \mathbb{C}^{R_1 \times R_2}$ , denominada matriz núcleo da decomposição PARATUCK2 que contém o grau de interação entre a  $r_1$ -ésima coluna de  $\mathbf{A}$  e a  $r_2$ -ésima coluna de  $\mathbf{B}$ .

Da mesma forma que a decomposição PARAFAC uma das representações mais importantes da decomposição PARATUCK2 é descrita utilizando os *slices* frontais do tensor  $\mathcal{X}$  que são obtidos a partir do seu fatiamento, ou seja, fixando  $i_3 = 1, \dots, I_3$  e variando os índices  $i_1$  e  $i_2$ . Dessa forma, a decomposição PARATUCK2 do  $i_3$ -ésimo *slice* frontal do tensor  $\mathcal{X}$  é dada por

$$\mathbf{X}_{\cdot i_3} = \mathbf{A} \mathbf{D}_{i_3}(\mathbf{C}^A) \mathbf{G} \mathbf{D}_{i_3}(\mathbf{C}^B) \mathbf{B}^T, \quad i_3 = 1, \dots, I_3. \quad (2.64)$$

em que  $\mathbf{D}_{i_3}(\mathbf{C}^A)$  e  $\mathbf{D}_{i_3}(\mathbf{C}^B)$  são matrizes diagonais formadas pela  $i_1$ -ésima linha de  $\mathbf{A}$  e pela  $i_2$ -ésima linha de  $\mathbf{B}$ , respectivamente.

Como dito anteriormente por se tratar de uma decomposição híbrida oriunda dos modelos PARAFAC e Tucker2 a decomposição PARATUCK2 mantém algumas propriedades de unicidade da decomposição PARAFAC. Esta característica favorece sua aplicação em problemas relacionados a processamento de sinais e sistemas de comunicação, por exemplo, essa decomposição foi explorada em [52], [7] e [53]. Entretanto, ao contrário da decomposição PARAFAC a decomposição PARATUCK2 não apresenta um critério de unicidade bem estabelecido e em [51] foi provado que a unicidade é garantida quando  $\mathbf{C}^A = \mathbf{C}^B$  sujeito a  $R_1 = R_2$  e a matriz núcleo  $\mathbf{G}$  não possuir elementos nulos [9]. Porém em [52], condições de unicidade para casos em que  $R_1 \neq R_2$  são abordadas.

## 2.5 Estimação de DoA Utilizando Decomposições Tensoriais

A utilização de diferentes esquemas de diversidade além das espaciais e temporais permite que os dados recebidos sejam organizados de forma multidimensional, proporcionando maiores graus de liberdade durante a separação dos sinais e conseqüentemente durante a realização da estimação de parâmetros. Em virtude do alcance dessa estrutura multidimensional no processo de recepção, diversos algoritmos baseados em decomposições tensoriais foram introduzidos dando origem a um grande interesse em processamento de sinais, principalmente por apresentarem propriedades de identificabilidade e unicidade superiores aos métodos matriciais [54]. No contexto deste trabalho, uma breve compilação sobre a utilização de decomposições tensoriais e conceitos referentes a álgebra multilinear no processo de estimação de DoA será descrita a seguir tomando como referência trabalhos publicados nos últimos anos.

Em [55] foi proposta uma forma tensorial equivalente ao MUSIC denominada Tensor-MUSIC, que representa a primeira abordagem tensorial desse algoritmo obtida a partir de uma formulação tensorial do método proposto em [56]. Em relação ao algoritmo ESPRIT, uma versão tensorial do *Unitary* ESPRIT para fontes não circulares denominada NC *R-D Unitary* Tensor-ESPRIT foi proposta em [57]. Nesse método os benefícios encontrados no NC *Unitary* ESPRIT proposto em [58] e no *R-D Unitary* Tensor-ESPRIT proposto em [59] são obtidos simultaneamente.

No que diz respeito a utilização da decomposição PARAFAC em processamento de sinais, diversos trabalhos utilizando diferentes esquemas de diversidade foram desenvolvidos. Em [45] é considerado um sistema de comunicação DS-CDMA (do inglês, *Direct-Sequence Code-Division Multiple Access*) em que a diversidade do código é utilizada na formulação de uma estrutura tensorial que possibilita uma estimação conjunta da DoA, código e das sequências transmitidas pelas fontes utilizando o algoritmo COMFAC proposto em [60]. Já em [4] é utilizada a diversidade de multi-arranjos obtida a partir da propriedade de invariância múltipla em arranjos de sensores. Nesse caso, a defasagem sofrida por cada sub-arranjo em relação ao sub-arranjo de referência origina a terceira dimensão do modelo tornando possível de forma semelhante a [45] estimar a DoA, porém nesse caso foi utilizado o algoritmo TALS (do inglês, *Trilinear Alternating Least Squares*).

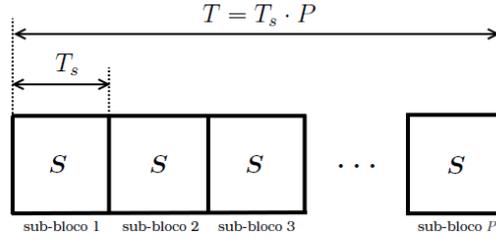
Outra forma de diversidade que vêm ganhando bastante atenção nos últimos anos é a diversidade de polarização. Esse tipo de diversidade é obtida quando arranjos de sensores vetoriais são utilizados no receptor. Em [61] e [62] sensores vetoriais são utilizados em arranjos lineares uniformes e em [63] são utilizados numa configuração *L-Shape*. Em ambos os casos as estimações de DoA são obtidas a partir da matriz de assinaturas espaciais de um modelo PARAFAC de terceira ordem. De forma semelhante, em [64] a estimação de DoA para fontes polarizadas é obtida a partir da decomposição PARAFAC de um tensor de covariância de quarta ordem utilizando o algoritmo QALS (do inglês, *Quadrilinear Alternative Least Squares*).

Na maioria dos trabalhos que utilizam decomposições tensoriais os algoritmos propostos para a realização da estimação de parâmetros são iterativos e baseados em ALS. Entretanto, outras formulações com soluções fechadas baseadas em LS-KRF (do inglês, *Least Squares Khatri-Rao Factorization*) vêm sendo utilizadas em sistemas de comunicação MIMO-OFDM [8] e em problemas de estimação de parâmetros *R*-dimensional [5]. Estes métodos apresentam baixa complexidade pois evitam a utilização de algoritmos iterativos.

## Métodos Tensoriais Propostos

Como descrito anteriormente, a utilização de diferentes esquemas de diversidade como código, multi-arranjos e polarização proporcionam um tratamento multidimensional dos dados em que a estimação de parâmetros é realizada utilizando decomposições tensoriais [45], [4] e [61]. Entretanto, estes não são os únicos esquemas de diversidade existentes, por exemplo, em [10] a diversidade em blocos de tempo foi utilizada na formulação de um receptor baseado na decomposição PARAFAC. Esse esquema de diversidade é obtido fixando-se a potência de cada fonte em cada sub-bloco de dados transmitido e alterando-a entre os diferentes sub-blocos, e não requer nenhuma sincronização precisa entre as fontes. Além disso, esta técnica pode ser facilmente implementada, pois necessita apenas de uma pequena variação de potência em relação a potência de transmissão padrão, que pode ser obtida de forma artificial sem nenhuma modificação a nível de *hardware* ou obtida naturalmente em canais com desvanecimento (sob a suposição de que a DoA das fontes variam lentamente em relação ao canal com desvanecimento) [10].

Entretanto, no contexto dos métodos tensoriais para estimação de DoA, em [10] a matriz de covariância dos sinais transmitidos é considerada perfeitamente conhecida e diagonal. Na prática, sabe-se que estas propriedades são válidas quando (i) os sinais provenientes das diferentes fontes são descorrelacionados e (ii) ocorre uma estimação perfeita da matriz de covariância espacial. Embora a primeira suposição possa ser válida em algumas situações, a segunda só será válida no caso assintótico e uma boa aproximação no cálculo da matriz de covariância requer um número suficientemente grande de *snapshots*. Em virtude disso, neste capítulo serão apresentados os métodos propostos neste trabalho. Estes métodos fornecem estimativas precisas da DoA sem contar com tais suposições idealizadas e resolvem de forma eficiente o problema de estimação de DoA sem recorrer à utilização de sequências de treinamento e sem conhecer a estrutura de



**Figura 3.1:** Estrutura de Transmissão dos Dados.

covariância das fontes, tornando-os mais atraentes em ambientes práticos em que a matriz de covariância é calculada a partir de um número finito de *snapshots*.

Inicialmente, receptores para os arranjos ULA e *2L-Shape* baseados nas decomposições PARATUCK2 e Tucker4 serão formulados e a partir dos modelos tensoriais obtidos, algoritmos iterativos baseados em ALS serão propostos para estimar a DoA das fontes. Em seguida, a diversidade multi-arranjos descrita em [4] será obtida utilizando a técnica *Spatial Smoothing* nos arranjos *2L-Shape* e Planar e as estimações da DoA serão obtidas utilizando os algoritmos ALS e LS-KRF. Em contraste a [10] que baseia-se no pressuposto da diagonalidade da matriz de covariância das fontes, os métodos propostos neste trabalho podem lidar eficientemente com estruturas de covariância não-diagonais e desconhecidas como será visto a seguir.

### 3.1 Receptor PARAFAC

Para descrever o receptor PARAFAC vamos considerar que um número fixo de  $T$  *snapshots* são divididos uniformemente em  $P$  sub-blocos, cada um composto por  $T_s = \frac{T}{P}$  *snapshots*. Em cada sub-bloco a  $m$ -ésima fonte transmite a sequência  $\mathbf{s}_m \in \mathbb{C}^{T_s \times 1}$  sendo garantido que a potência de cada forma de onda transmitida varia no  $p$ -ésimo sub-bloco. A Figura 3.1 ilustra a estrutura de transmissão dos dados. Vamos definir a matriz de potências  $\mathbf{W} = [\mathbf{w}_1, \dots, \mathbf{w}_M] \in \mathbb{R}^{P \times M}$  em que sua  $m$ -ésima coluna contém os coeficientes relacionados as potências de transmissão da  $m$ -ésima fonte no decorrer dos  $P$  sub-blocos e  $\mathbf{S} = [\mathbf{s}_1, \dots, \mathbf{s}_M]^T \in \mathbb{C}^{M \times T_s}$  como sendo a matriz que contém a concatenação das sequências transmitidas pelas  $M$  fontes. A partir dessas considerações a matriz dos dados recebidos no  $p$ -ésimo sub-bloco é dada por

$$\mathbf{X}_{(p)} = \mathbf{A}D_p(\mathbf{W})\mathbf{S} + \mathbf{V}_{(p)} \in \mathbb{C}^{N \times T_s}, \quad p = 1, \dots, P. \quad (3.1)$$

em que  $D_p(\mathbf{W}) \in \mathbb{R}^{M \times M}$  é uma matriz diagonal que contém as potências de transmissão das  $M$  fontes no  $p$ -ésimo sub-bloco. A matriz de covariância

especial dos dados recebidos no  $p$ -ésimo sub-bloco  $\mathbf{R}_{(p)} \in \mathbb{C}^{N \times N}$  é obtida da seguinte forma

$$\begin{aligned} \mathbf{R}_{(p)} &\triangleq \frac{1}{T_s} \mathbf{X}_{(p)} \mathbf{X}_{(p)}^H = \mathbf{A} D_p(\mathbf{W}) \underbrace{\left( \frac{1}{T_s} \mathbf{S} \mathbf{S}^H \right)}_{\mathbf{R}_s} D_p(\mathbf{W}^*) \mathbf{A}^H + \underbrace{\frac{1}{T_s} \mathbf{V}_{(p)} \mathbf{V}_{(p)}^H}_{\sigma_v^2 \mathbf{I}}, \\ \mathbf{R}_{(p)} &= \mathbf{A} D_p(\mathbf{W}) \mathbf{R}_s D_p(\mathbf{W}^*) \mathbf{A}^H + \sigma_v^2 \mathbf{I} \in \mathbb{C}^{N \times N}, \end{aligned} \quad (3.2)$$

subtraindo por simplicidade de notação  $\bar{\mathbf{R}}_{(p)} = \mathbf{R}_{(p)} - \sigma_v^2 \mathbf{I}$  podemos reescrever a equação (3.2) como

$$\bar{\mathbf{R}}_{(p)} = \mathbf{A} D_p(\mathbf{W}) \mathbf{R}_s D_p(\mathbf{W}^*) \mathbf{A}^H. \quad (3.3)$$

A partir da equação (3.3) é possível perceber que a técnica apresentada em [10] adiciona uma dimensão temporal no processamento dos sinais, possibilitando que as múltiplas matrizes de covariâncias dos dados recebidos no decorrer dos  $T$  *snapshots* sejam organizadas de forma tensorial. Tal organização possibilita que a estimação da matriz de assinaturas espaciais e conseqüentemente da DoA das fontes seja realizada utilizando técnicas baseadas em decomposições tensoriais. Nesta seção, será apresentada uma versão modificada do método baseado na decomposição PARAFAC proposto em [10]. Esta versão diferentemente da descrita em [10] possibilita realizar a estimação conjunta das matrizes  $\mathbf{A}$  e  $\mathbf{R}_s$ . Para descrevê-la, vamos inicialmente reescrever a equação (3.3) da seguinte forma

$$\begin{aligned} \bar{\mathbf{R}}_{(p)} &= \mathbf{A} \underbrace{D_p(\mathbf{W}) D_p(\mathbf{W}^*)}_{D_p(\bar{\mathbf{W}})} \mathbf{R}_s \mathbf{A}^H, \text{ onde } \bar{\mathbf{W}} = \mathbf{W} \odot \mathbf{W}^*. \\ \bar{\mathbf{R}}_{(p)} &= \mathbf{A} D_p(\bar{\mathbf{W}}) \mathbf{R}_s \mathbf{A}^H \rightarrow \bar{\mathbf{R}}_{(p)} = \mathbf{A} D_p(\underbrace{\bar{\mathbf{W}} \mathbf{R}_s}_{\Omega}) \mathbf{A}^H, \\ \bar{\mathbf{R}}_{(p)} &= \mathbf{A} D_p(\Omega) \mathbf{A}^H \in \mathbb{C}^{N \times N}. \end{aligned} \quad (3.4)$$

A equação (3.4) descreve a decomposição PARAFAC [9] do tensor  $\bar{\mathcal{R}} \in \mathbb{C}^{N \times N \times P}$ , obtida através do cálculo das múltiplas covariâncias espaciais dos dados recebidos no arranjo. Nesse caso, as seguintes correspondências com a equação (2.44) são obtidas:

$$\begin{aligned} (I_1, I_2, I_3, R) &\leftrightarrow (N, N, P, M) \\ (\mathbf{A}, \mathbf{B}, \mathbf{C}) &\leftrightarrow (\mathbf{A}, \mathbf{A}^*, \Omega). \end{aligned} \quad (3.5)$$

A partir da equação (3.4) é possível descrever as matrizes *slices* referentes as diferentes dimensões do tensor  $\bar{\mathcal{R}}$ . De acordo com [33] e [6] obtemos a seguinte

notação para a decomposição PARAFAC:

$$\begin{aligned}\bar{\mathbf{R}}_{n..} &= \mathbf{A}^* D_n(\mathbf{A}) \Omega^T, \quad n = 1, \dots, N. \\ \bar{\mathbf{R}}_{.n.} &= \Omega D_n(\mathbf{A}^*) \mathbf{A}^T, \quad n = 1, \dots, N. \\ \bar{\mathbf{R}}_{..p} &= \mathbf{A} D_p(\Omega) (\mathbf{A}^*)^T, \quad p = 1, \dots, P.\end{aligned}\tag{3.6}$$

As formas matriciadas de  $\bar{\mathcal{R}}$  são obtidas através da concatenação das diferentes matrizes *slices* conforme descrito a seguir

$$\begin{aligned}[\bar{\mathcal{R}}]_{(1)} &= \begin{bmatrix} \bar{\mathbf{R}}_{..1} & \cdots & \bar{\mathbf{R}}_{..P} \end{bmatrix} = \mathbf{A} \begin{bmatrix} D_1(\Omega) (\mathbf{A}^*)^T & \cdots & D_P(\Omega) (\mathbf{A}^*)^T \end{bmatrix} \\ [\bar{\mathcal{R}}]_{(1)} &= \mathbf{A} (\Omega \diamond \mathbf{A}^*)^T.\end{aligned}\tag{3.7}$$

$$\begin{aligned}[\bar{\mathcal{R}}]_{(2)} &= \begin{bmatrix} \bar{\mathbf{R}}_{.1.}^T & \cdots & \bar{\mathbf{R}}_{.P.}^T \end{bmatrix} = \mathbf{A}^* \begin{bmatrix} D_1(\Omega) \mathbf{A}^T & \cdots & D_P(\Omega) \mathbf{A}^T \end{bmatrix} \\ [\bar{\mathcal{R}}]_{(2)} &= \mathbf{A}^* (\Omega \diamond \mathbf{A})^T.\end{aligned}\tag{3.8}$$

$$\begin{aligned}[\bar{\mathcal{R}}]_{(3)} &= \begin{bmatrix} \bar{\mathbf{R}}_{.1.} & \cdots & \bar{\mathbf{R}}_{.N.} \end{bmatrix} = \Omega \begin{bmatrix} D_1(\mathbf{A}^*) \mathbf{A}^T & \cdots & D_N(\mathbf{A}^*) \mathbf{A}^T \end{bmatrix} \\ [\bar{\mathcal{R}}]_{(3)} &= \Omega (\mathbf{A}^* \diamond \mathbf{A})^T.\end{aligned}\tag{3.9}$$

em que  $[\bar{\mathcal{R}}]_{(1)} \in \mathbb{C}^{N \times NP}$ ,  $[\bar{\mathcal{R}}]_{(2)} \in \mathbb{C}^{N \times NP}$  e  $[\bar{\mathcal{R}}]_{(3)} \in \mathbb{C}^{P \times N^2}$ , respectivamente. Como  $\Omega = \bar{\mathbf{W}} \mathbf{R}_s$  podemos reescrever a equação (3.9) como

$$[\bar{\mathcal{R}}]_{(3)} = \bar{\mathbf{W}} \mathbf{R}_s (\mathbf{A}^* \diamond \mathbf{A})^T.\tag{3.10}$$

Utilizando a seguinte propriedade do produto de Khatri-Rao [65]

$$\text{vec}(\mathbf{ABC}^T) = (\mathbf{C} \diamond \mathbf{A}) \text{vecd}(\mathbf{B}),\tag{3.11}$$

em que  $\mathbf{B}$  é uma matriz diagonal e  $\mathbf{A}$ ,  $\mathbf{B}$  e  $\mathbf{C}$  possuem dimensões compatíveis. O operador  $\text{vecd}(\mathbf{B})$  converte os elementos da diagonal principal de  $\mathbf{B}$  num vetor. Podemos reescrever a equação (3.10) da seguinte forma

$$\text{vec}([\bar{\mathcal{R}}]_{(3)}) = (\mathbf{A}^* \diamond \mathbf{A} \diamond \bar{\mathbf{W}}) \text{vecd}(\mathbf{R}_s).\tag{3.12}$$

A proposta do método PARAFAC é realizar a estimação da matriz de assinaturas espaciais  $\mathbf{A}$  e conseqüentemente da DoA dos  $M$  sinais incidentes no arranjo assumindo que a matriz de potências  $\mathbf{W}$  é conhecida no receptor. Mais especificamente o objetivo desse método é minimizar a seguinte função custo:

$$f_{\text{custo}}(\hat{\mathbf{A}}, \hat{\mathbf{R}}_s) = \left\| [\bar{\mathcal{R}}]_{(1)} - \hat{\mathbf{A}} \left( \bar{\mathbf{W}} \hat{\mathbf{R}}_s \diamond \hat{\mathbf{A}}^* \right)^T \right\|_F^2.\tag{3.13}$$

Sendo esta uma função não-linear nos parâmetros a serem estimados uma solução é obtida utilizando um método iterativo baseado em mínimos quadrados alternados (do inglês, ALS - *Alternating Least Squares*) [66]. O algoritmo ALS-PARAFAC modificado em relação a [10] consiste na realização das estimações das matrizes desconhecidas. Em cada etapa do algoritmo a estimativa de cada matriz é obtida condicionada aos valores estimados em iterações anteriores. O algoritmo ALS-PARAFAC utilizado na estimação da matriz de assinaturas espaciais será apresentado detalhadamente a seguir:

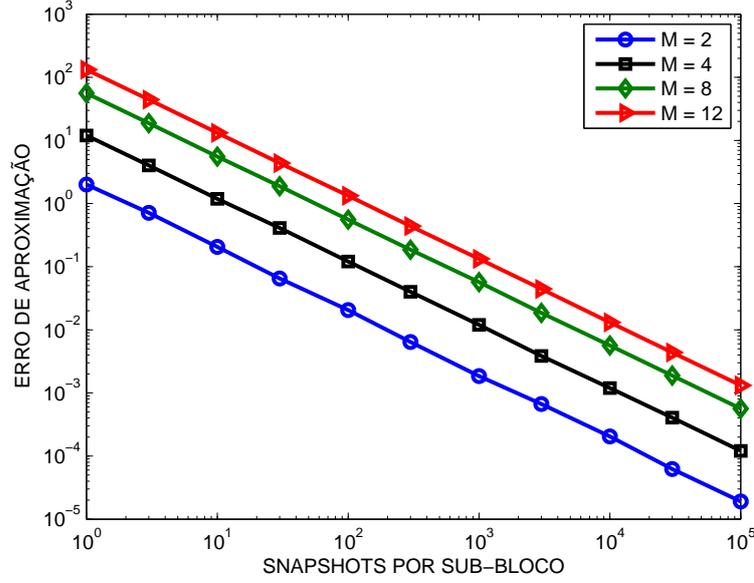
### Algoritmo ALS-PARAFAC

1. Fixe  $i = 0$ ;  
Inicialize aleatoriamente  $\hat{\mathbf{R}}_s^{(i=0)}$ ,  $\hat{\mathbf{A}}_{(i=0)}$ ,  $\hat{\mathbf{B}}_{(i=0)} = \hat{\mathbf{A}}_{(i=0)}^*$ ;
2.  $i = i + 1$ ;
3. A partir de  $\bar{\mathbf{W}}$  e  $\hat{\mathbf{R}}_s^{(i-1)}$  calcule  $\hat{\mathbf{C}}_{(i)} = \bar{\mathbf{W}} \hat{\mathbf{R}}_s^{(i-1)}$ ;
4. Utilizando  $[\bar{\mathcal{R}}]_{(1)}$  calcule a estimação LS de  $\hat{\mathbf{A}}_{(i)}$ :  
$$\hat{\mathbf{A}}_{(i)} = [\bar{\mathcal{R}}]_{(1)} \left[ \left( \hat{\mathbf{C}}_{(i)} \diamond \hat{\mathbf{B}}_{(i-1)} \right)^{\top} \right]^{\dagger};$$
5. Utilizando  $[\bar{\mathcal{R}}]_{(2)}$  calcule a estimação LS de  $\hat{\mathbf{B}}_{(i)}$ :  
$$\hat{\mathbf{B}}_{(i)} = [\bar{\mathcal{R}}]_{(2)} \left[ \left( \hat{\mathbf{C}}_{(i)} \diamond \hat{\mathbf{A}}_{(i)} \right)^{\top} \right]^{\dagger};$$
6. Utilizando  $\text{vec}([\bar{\mathcal{R}}]_{(3)})$  calcule a estimação LS de  $\text{vecd}(\hat{\mathbf{R}}_s^{(i)})$ :  
$$\text{vecd}(\hat{\mathbf{R}}_s^{(i)}) = \left( \hat{\mathbf{B}}_{(i)} \diamond \hat{\mathbf{A}}_{(i)} \diamond \bar{\mathbf{W}} \right)^{\dagger} \text{vec}([\bar{\mathcal{R}}]_{(3)});$$
  
$$\hat{\mathbf{R}}_s^{(i)} = \text{diag}(\text{vecd}(\hat{\mathbf{R}}_s^{(i)}));$$
7. Repetir os passos 2-6 até a convergência.
8. Após a convergência calcule  $\hat{\mathbf{A}}_{(\text{final})} = \frac{\hat{\mathbf{A}}_{(i)} + \hat{\mathbf{B}}_{(i)}^*}{2}$ .

A convergência do algoritmo ALS-PARAFAC na  $i$ -ésima iteração é alcançada quando a diferença entre o tensor original e sua versão reconstruída utilizando as matrizes estimadas não varia consideravelmente entre as iterações  $i - 1$  e  $i$ . Mais detalhadamente, definindo-se o erro residual na  $i$ -ésima iteração como:

$$e_{(i)} = \left\| [\bar{\mathcal{R}}]_{(1)} - \hat{\mathbf{A}} \left( \bar{\mathbf{W}} \hat{\mathbf{R}}_s \diamond \hat{\mathbf{A}}^* \right)^{\top} \right\|_F^2, \quad (3.14)$$

considera-se que o algoritmo converge na  $i$ -ésima iteração se  $|e_{(i)} - e_{(i-1)}| \leq 10^{-6}$ . Note que o algoritmo apresentado não aborda a característica de simetria descrita na equação (3.4) em que  $\mathbf{B} = \mathbf{A}^*$ , ou seja, as matrizes  $\hat{\mathbf{A}}$  e  $\hat{\mathbf{B}}$  são tratadas como variáveis independentes e a simetria do modelo é abordada somente no cálculo de  $\hat{\mathbf{A}}_{(\text{final})}$ .



**Figura 3.2:** Erro de Aproximação vs. *Snapshots* por Sub-Bloco.

### 3.2 Receptor PARATUCK2

A partir do desenvolvimento matemático do receptor PARAFAC é possível perceber que a equação (3.12) possui uma solução ideal quando a matriz de covariância  $\mathbf{R}_s$  for uma matriz diagonal. Entretanto, para que esta característica seja alcançada é necessário garantir que as sequências transmitidas em cada sub-bloco sejam descorrelacionados, de forma que  $\mathbf{R}_s$  será uma matriz diagonal exata no caso assintótico em que  $T_s \rightarrow \infty$ , ou seja,

$$\frac{1}{T_s} \mathbf{S} \mathbf{S}^H \rightarrow E \{ \mathbf{s}(t) \mathbf{s}^H(t) \} = \text{diag}(\varphi_1, \dots, \varphi_M), \quad m = 1, \dots, M. \quad (3.15)$$

em que  $\varphi_m$  é a variância da  $m$ -ésima fonte e não são necessariamente iguais para  $m = 1, \dots, M$ . Nos casos em que as covariâncias são calculadas sobre um intervalo finito de *snapshots* a diagonalidade da matriz de covariância dos sinais transmitidos não se sustenta e  $\mathbf{R}_s = \frac{1}{T_s} \mathbf{S} \mathbf{S}^H \neq \text{diag}(\varphi_1, \dots, \varphi_M)$  mesmo para fontes descorrelacionadas. Nesse caso, ocorrerá um erro de aproximação no cálculo da covariância das fontes definido por  $\epsilon = \|\mathbf{R}_s - \mathbf{I}\|_F^2$  que acarreta um erro de modelagem do receptor PARAFAC, tornando o método proposto em [10] inviável nessas condições. O erro de aproximação  $\epsilon$  em função do número de *snapshots* por sub-bloco considerando que  $\varphi_m = 1$  para  $m = 1, \dots, M$  é ilustrado na Figura 3.2.

Além disso, quando um número elevado de *snapshots* é considerado a utilização da decomposição PARAFAC diretamente no tensor de dados, conforme a equação (3.1), torna-se inviável em virtude da complexidade

computacional oriunda do número elevado de dados coletados. Em virtude dessas deficiências, é desejável utilizar um método capaz de realizar de forma adequada a estimação da DoA dos  $M$  sinais incidentes descartando suposições idealizadas conforme [10] e que possua uma complexidade computacional independente de  $T_s$ . Dessa forma, negligenciando o papel do ruído aditivo proveniente do processo de recepção dos sinais nos sensores que compõem o arranjo e supondo que  $\mathbf{R}_s$  possui uma estrutura arbitrária não necessariamente diagonal, as múltiplas matrizes de covariâncias espaciais dos dados recebidos são descritas conforme a equação (3.3):

$$\bar{\mathbf{R}}_{(p)} = \mathbf{A}D_p(\mathbf{W})\mathbf{R}_sD_p(\mathbf{W}^*)\mathbf{A}^H, \quad p = 1, \dots, P.$$

e representam os *slices* frontais do tensor  $\bar{\mathcal{R}} \in \mathbb{C}^{N \times N \times P}$  descrito por uma decomposição PARATUCK2 simétrica [32]. Nesse caso, as seguintes correspondências com a equação (2.64) são obtidas:

$$\begin{aligned} (I_1, I_2, I_3, R_1, R_2) &\leftrightarrow (N, N, P, M, M) \\ (\mathbf{A}, \mathbf{C}^A, \mathbf{G}, \mathbf{C}^B, \mathbf{B}) &\leftrightarrow (\mathbf{A}, \mathbf{W}, \mathbf{R}_s, \mathbf{W}^*, \mathbf{A}^*). \end{aligned} \quad (3.16)$$

Para iniciar a descrição do receptor PARATUCK2 vamos definir

$$\bar{\mathbf{r}}_{(p)} = \text{vec}(\bar{\mathbf{R}}_{(p)}) \in \mathbb{C}^{N^2 \times 1}, \quad p = 1, \dots, P. \quad (3.17)$$

e utilizando a seguinte propriedade do produto de Kronecker [65]

$$\text{vec}(\mathbf{ABC}^T) = (\mathbf{C} \otimes \mathbf{A})\text{vec}(\mathbf{B}), \quad (3.18)$$

em que  $\mathbf{A}, \mathbf{B}$  e  $\mathbf{C}$  são matrizes com dimensões compatíveis, podemos reescrever a equação (3.17) da seguinte forma

$$\bar{\mathbf{r}}_{(p)} = (\mathbf{A}^* \otimes \mathbf{A})\text{vec}(D_p(\mathbf{W})\mathbf{R}_sD_p(\mathbf{W}^*)), \quad (3.19)$$

aplicando a mesma propriedade no segundo termo do lado direito da equação (3.19) obtemos

$$\bar{\mathbf{r}}_{(p)} = (\mathbf{A}^* \otimes \mathbf{A})(D_p(\mathbf{W}^*) \otimes D_p(\mathbf{W}))\text{vec}(\mathbf{R}_s). \quad (3.20)$$

Note que  $D_p(\mathbf{W}^*) \otimes D_p(\mathbf{W}) \in \mathbb{R}^{M^2 \times M^2}$  é uma matriz diagonal e pode ser reescrita de forma equivalente como

$$D_p(\mathbf{W}^*) \otimes D_p(\mathbf{W}) = \text{diag}(\mathbf{W}_p^H \otimes \mathbf{W}_p^T),$$

utilizando o fato de que  $\text{diag}(\mathbf{a})\mathbf{b} = \text{diag}(\mathbf{b})\mathbf{a}$ , em que  $\mathbf{a}$  e  $\mathbf{b}$  são vetores com dimensões compatíveis, podemos reescrever a equação (3.20) da seguinte forma

$$\bar{\mathbf{r}}_{(p)} = (\mathbf{A}^* \otimes \mathbf{A}) \text{diag}(\text{vec}(\mathbf{R}_s)) (\mathbf{W}_p^H \otimes \mathbf{W}_p^T). \quad (3.21)$$

Uma das formas matriciadas do tensor  $\bar{\mathcal{R}} \in \mathbb{C}^{N \times N \times P}$  pode ser obtida através da concatenação coluna a coluna dos vetores descritos em (3.21), ou seja, para  $p = 1, \dots, P$  obtemos

$$\begin{aligned} [\bar{\mathcal{R}}]_{(1)} &= \begin{bmatrix} \bar{\mathbf{r}}_{(1)} & \cdots & \bar{\mathbf{r}}_{(P)} \end{bmatrix} \in \mathbb{C}^{N^2 \times P} \\ &= \begin{bmatrix} (\mathbf{A}^* \otimes \mathbf{A}) \text{diag}(\text{vec}(\mathbf{R}_s)) (\mathbf{W}_1^H \otimes \mathbf{W}_1^T) & \cdots & (\mathbf{A}^* \otimes \mathbf{A}) \text{diag}(\text{vec}(\mathbf{R}_s)) (\mathbf{W}_P^H \otimes \mathbf{W}_P^T) \end{bmatrix} \\ &= (\mathbf{A}^* \otimes \mathbf{A}) \text{diag}(\text{vec}(\mathbf{R}_s)) \begin{bmatrix} (\mathbf{W}_1^H \otimes \mathbf{W}_1^T) & \cdots & (\mathbf{W}_P^H \otimes \mathbf{W}_P^T) \end{bmatrix} \\ [\bar{\mathcal{R}}]_{(1)} &= (\mathbf{A}^* \otimes \mathbf{A}) \text{diag}(\text{vec}(\mathbf{R}_s)) (\mathbf{W}^H \diamond \mathbf{W}^T), \end{aligned} \quad (3.22)$$

utilizando  $\text{vec}(\mathbf{ABC}) = (\mathbf{C}^T \diamond \mathbf{A}) \text{vecd}(\mathbf{B})$ , em que  $\mathbf{B}$  é uma matriz diagonal, podemos reescrever a equação (3.22) na sua forma vetorial equivalente

$$\begin{aligned} \text{vec}([\bar{\mathcal{R}}]_{(1)}) &= \left[ (\mathbf{W}^H \diamond \mathbf{W}^T)^T \diamond (\mathbf{A}^* \otimes \mathbf{A}) \right] \text{vecd}(\text{diag}(\text{vec}(\mathbf{R}_s))), \\ \text{vec}([\bar{\mathcal{R}}]_{(1)}) &= \left[ (\mathbf{W}^H \diamond \mathbf{W}^T)^T \diamond (\mathbf{A}^* \otimes \mathbf{A}) \right] \text{vec}(\mathbf{R}_s). \end{aligned} \quad (3.23)$$

Dando continuidade na descrição do receptor PARATUCK2 é necessário determinar as outras duas formas matriciadas de  $\bar{\mathcal{R}}$  definidas a seguir:

$$[\bar{\mathcal{R}}]_{(2)} = \begin{bmatrix} \bar{\mathbf{R}}_{(1)} \\ \vdots \\ \bar{\mathbf{R}}_{(P)} \end{bmatrix} \in \mathbb{C}^{NP \times N}, \quad [\bar{\mathcal{R}}]_{(3)} = \begin{bmatrix} \bar{\mathbf{R}}_{(1)}^T \\ \vdots \\ \bar{\mathbf{R}}_{(P)}^T \end{bmatrix} \in \mathbb{C}^{NP \times N}, \quad (3.24)$$

definindo também,

$$\mathbf{F}_{(p)} = D_p(\mathbf{W}) \mathbf{R}_s D_p(\mathbf{W}^*), \quad p = 1, \dots, P. \quad (3.25)$$

temos que

$$\begin{aligned} [\bar{\mathcal{R}}]_{(2)} &= \begin{bmatrix} \mathbf{A} \mathbf{F}_{(1)} \mathbf{A}^H \\ \vdots \\ \mathbf{A} \mathbf{F}_{(P)} \mathbf{A}^H \end{bmatrix} = \begin{bmatrix} \mathbf{A} \mathbf{F}_{(1)} \\ \vdots \\ \mathbf{A} \mathbf{F}_{(P)} \end{bmatrix} \mathbf{A}^H = \underbrace{\begin{bmatrix} \mathbf{A} & & \\ & \ddots & \\ & & \mathbf{A} \end{bmatrix}}_{P \text{ vezes}} \begin{bmatrix} \mathbf{F}_{(1)} \\ \vdots \\ \mathbf{F}_{(P)} \end{bmatrix} \mathbf{A}^H, \\ [\bar{\mathcal{R}}]_{(2)} &= (\mathbf{I}_P \otimes \mathbf{A}) \mathbf{F}_2 \mathbf{A}^H, \end{aligned} \quad (3.26)$$

e

$$\begin{aligned}
[\bar{\mathcal{R}}]_{(3)} &= \begin{bmatrix} \mathbf{A}^* \mathbf{F}_{(1)}^T \mathbf{A}^T \\ \vdots \\ \mathbf{A}^* \mathbf{F}_{(P)}^T \mathbf{A}^T \end{bmatrix} = \begin{bmatrix} \mathbf{A}^* \mathbf{F}_{(1)}^T \\ \vdots \\ \mathbf{A}^* \mathbf{F}_{(P)}^T \end{bmatrix} \mathbf{A}^T = \underbrace{\begin{bmatrix} \mathbf{A}^* & & \\ & \ddots & \\ & & \mathbf{A}^* \end{bmatrix}}_{P \text{ vezes}} \begin{bmatrix} \mathbf{F}_{(1)}^T \\ \vdots \\ \mathbf{F}_{(P)}^T \end{bmatrix} \mathbf{A}^T, \\
[\bar{\mathcal{R}}]_{(3)} &= (\mathbf{I}_P \otimes \mathbf{A}^*) \mathbf{F}_3 \mathbf{A}^T, \tag{3.27}
\end{aligned}$$

em que

$$\mathbf{F}_2 = \begin{bmatrix} \mathbf{F}_{(1)} \\ \vdots \\ \mathbf{F}_{(P)} \end{bmatrix} \in \mathbb{C}^{PM \times M}, \quad \mathbf{F}_3 = \begin{bmatrix} \mathbf{F}_{(1)}^T \\ \vdots \\ \mathbf{F}_{(P)}^T \end{bmatrix} \in \mathbb{C}^{PM \times M}. \tag{3.28}$$

Fazendo uso das formas matriciadas do tensor  $\bar{\mathcal{R}}$  descritas nas equações (3.23), (3.26) e (3.27) um algoritmo iterativo baseado no método ALS será formulado. Para o algoritmo ALS-PARATUCK2 também é assumido que a matriz de potências  $\mathbf{W}$  é conhecida no receptor e seu objetivo é realizar a estimação da matriz de covariância das sequências transmitidas  $\mathbf{R}_s$  e da matriz de assinaturas espaciais  $\mathbf{A}$  condicionadas a minimização da seguinte função custo:

$$f_{\text{custo}}(\hat{\mathbf{A}}, \hat{\mathbf{R}}_s) = \sum_{p=1}^P \left\| \bar{\mathcal{R}}_{(p)} - \hat{\mathbf{A}} D_p(\mathbf{W}) \hat{\mathbf{R}}_s D_p(\mathbf{W}^*) \hat{\mathbf{A}}^H \right\|_F^2. \tag{3.29}$$

As etapas do algoritmo ALS-PARATUCK2 serão descritas detalhadamente no escopo apresentado a seguir:

### Algoritmo ALS-PARATUCK2

- 
1. Fixe  $i = 0$ ;  
Inicialize aleatoriamente  $\hat{\mathbf{G}}_{(i=0)}$  e  $\hat{\mathbf{A}}_{(i=0)}$ ;
  2.  $i = i + 1$ ;
  3. A partir de  $\hat{\mathbf{G}}_{(i-1)}$  calcule  $\hat{\mathbf{F}}_2^{(i)}$  e  $\hat{\mathbf{F}}_3^{(i)}$  respectivamente de acordo com as equações (3.25) e (3.28);
  4. Utilizando  $[\bar{\mathcal{R}}]_{(2)}$  calcule a estimação LS de  $\hat{\mathbf{B}}_{(i)}$ :  

$$\hat{\mathbf{B}}_{(i)} = \left[ \left[ (\mathbf{I}_P \otimes \hat{\mathbf{A}}_{(i-1)}) \hat{\mathbf{F}}_2^{(i)} \right]^\dagger [\bar{\mathcal{R}}]_{(2)} \right]^T;$$
  5. Utilizando  $[\bar{\mathcal{R}}]_{(3)}$  calcule a estimação LS de  $\hat{\mathbf{A}}_{(i)}$ :  

$$\hat{\mathbf{A}}_{(i)} = \left[ \left[ (\mathbf{I}_P \otimes \hat{\mathbf{B}}_{(i)}) \hat{\mathbf{F}}_3^{(i)} \right]^\dagger [\bar{\mathcal{R}}]_{(3)} \right]^T;$$
  6. Utilizando  $[\bar{\mathcal{R}}]_{(1)}$  calcule a estimação LS de  $\hat{\mathbf{G}}_{(i)}$ ;

$$\text{vec} \left( \hat{\mathbf{G}}_{(i)} \right) = \left[ (\mathbf{W}^H \diamond \mathbf{W}^T)^T \diamond \left( \hat{\mathbf{B}}_{(i)} \otimes \hat{\mathbf{A}}_{(i)} \right) \right]^\dagger \text{vec} \left( [\bar{\mathbf{R}}]_{(1)} \right);$$

$$\hat{\mathbf{G}}_{(i)} = \text{unvec}_{M \times M} \left( \text{vec} \left( \hat{\mathbf{G}}_{(i)} \right) \right);$$

7. Repetir os passos 2-6 até a convergência.

8. Após a convergência calcule  $\hat{\mathbf{A}}_{(\text{final})} = \frac{\hat{\mathbf{A}}_{(i)} + \hat{\mathbf{B}}_{(i)}^*}{2}$ .

Para o algoritmo ALS-PARATUCK2 o erro residual entre o tensor recebido no arranjo de sensores e sua versão estimada na  $i$ -ésima iteração é dado por

$$e_{(i)} = \sum_{p=1}^P \left\| \bar{\mathbf{R}}_{(p)} - \hat{\mathbf{A}} D_p(\mathbf{W}) \hat{\mathbf{R}}_s D_p(\mathbf{W}^*) \hat{\mathbf{A}}^H \right\|_F^2, \quad (3.30)$$

e sua convergência na  $i$ -ésima iteração também é alcançada quando a diferença  $|e_{(i)} - e_{(i-1)}| \leq 10^{-6}$ , garantindo que o erro residual entre duas iterações sucessivas não varia consideravelmente.

Note que no estimador PARATUCK2 é assumido naturalmente que a matriz  $\mathbf{R}_s \neq \text{diag}(\varphi_1, \dots, \varphi_M)$ , ou seja, a estimação de  $\mathbf{A}$  pode ser realizada utilizando um número finito de *snapshots*. Este fato acarreta um tempo de observação menor, sendo mais razoável nesse caso, assumir que o tempo de coerência do canal é maior do que o intervalo de observação e que  $\mathbf{A}$  é invariante durante este intervalo. Além disso, a utilização do estimador PARATUCK2 possibilita uma estimação adequada quando os usuários são correlacionados, situação em que o método proposto em [10] não se aplica.

### 3.3 Receptor Tucker4

Um terceiro método para solucionar o problema de estimação da matriz de assinaturas espaciais será apresentado nesta seção. Diferentemente dos receptores PARAFAC e PARATUCK2 que utilizam um total de  $P$  matrizes de covariâncias espaciais calculadas a partir dos dados recebidos em cada sub-bloco, o presente método é baseado no cálculo de apenas uma matriz de covariância espaço-temporal multimodo desses mesmos dados. Para descrevê-lo devemos inicialmente realizar uma coleta dos dados recebidos no decorrer dos  $P$  sub-blocos descritos na equação (3.1), obtendo-se uma única matriz resultante descrita da seguinte forma

$$\underbrace{\begin{bmatrix} \mathbf{X}_{(1)} \\ \mathbf{X}_{(2)} \\ \vdots \\ \mathbf{X}_{(P)} \end{bmatrix}}_U = \begin{bmatrix} AD_1(\mathbf{W}) \\ AD_2(\mathbf{W}) \\ \vdots \\ AD_P(\mathbf{W}) \end{bmatrix} \mathbf{S} + \underbrace{\begin{bmatrix} \mathbf{V}_{(1)} \\ \mathbf{V}_{(2)} \\ \vdots \\ \mathbf{V}_{(P)} \end{bmatrix}}_V \in \mathbb{C}^{NP \times T_s}, \quad (3.31)$$

ou de forma equivalente

$$\mathbf{U} = (\mathbf{W} \diamond \mathbf{A}) \mathbf{S} + \mathbf{V} \in \mathbb{C}^{NP \times T_s}. \quad (3.32)$$

Vamos introduzir a matriz de covariância espaço-temporal multimodo referente aos dados recebidos no decorrer dos  $P$  sub-blocos, dada por

$$\begin{aligned} \mathbf{R}_{\text{mm}} &\triangleq \frac{1}{T_s} \mathbf{U} \mathbf{U}^H = (\mathbf{W} \diamond \mathbf{A}) \underbrace{\left( \frac{1}{T_s} \mathbf{S} \mathbf{S}^H \right)}_{\mathbf{R}_s} (\mathbf{W} \diamond \mathbf{A})^H + \underbrace{\frac{1}{T_s} \mathbf{V} \mathbf{V}^H}_{\sigma_v^2 \mathbf{I}}, \\ \mathbf{R}_{\text{mm}} &= (\mathbf{W} \diamond \mathbf{A}) \mathbf{R}_s (\mathbf{W} \diamond \mathbf{A})^H + \sigma_v^2 \mathbf{I} \in \mathbb{C}^{NP \times NP}. \end{aligned} \quad (3.33)$$

Realizando a subtração da estimação da potência do ruído de  $\mathbf{R}_{\text{mm}}$ , ou seja,  $\bar{\mathbf{R}}_{\text{mm}} = \mathbf{R}_{\text{mm}} - \sigma_v^2 \mathbf{I}$ , obtemos

$$\bar{\mathbf{R}}_{\text{mm}} = (\mathbf{W} \diamond \mathbf{A}) \mathbf{R}_s (\mathbf{W} \diamond \mathbf{A})^H \in \mathbb{C}^{NP \times NP}. \quad (3.34)$$

Um caso especial da relação existente entre os produtos de Kronecker e Khatri-Rao ocorre quando duas matrizes quaisquer  $\mathbf{A}$  e  $\mathbf{B}$  possuem o mesmo número de colunas e não necessariamente o mesmo número de linhas, ou seja,  $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_n]$  e  $\mathbf{B} = [\mathbf{b}_1, \dots, \mathbf{b}_n]$  em que  $\mathbf{a}_i$  e  $\mathbf{b}_i$  representam a  $i$ -ésima coluna de  $\mathbf{A}$  e  $\mathbf{B}$ , respectivamente. Nesse caso, o produto de Khatri-Rao pode ser interpretado como um caso especial do produto de Kronecker e a seguinte propriedade é válida [65]

$$\mathbf{A} \diamond \mathbf{B} = (\mathbf{A} \otimes \mathbf{B}) \mathbf{L}_1, \quad (3.35)$$

em que  $\mathbf{L}_1$  é uma matriz de seleção de ordem  $n^2 \times n$ . A partir da equação (3.35) podemos reescrever a equação (3.34) da seguinte forma

$$\begin{aligned} \underbrace{\bar{\mathbf{R}}_{\text{mm}}}_{[\bar{\mathcal{R}}]_{(1,2);(3,4)}} &= (\mathbf{W} \otimes \mathbf{A}) \underbrace{\mathbf{L}_1 \mathbf{R}_s \mathbf{L}_1^H}_{[\mathcal{R}_s]_{(1,2);(3,4)}} (\mathbf{W} \otimes \mathbf{A})^H, \\ [\bar{\mathcal{R}}]_{(1,2);(3,4)} &= (\mathbf{W} \otimes \mathbf{A}) [\mathcal{R}_s]_{(1,2);(3,4)} (\mathbf{W} \otimes \mathbf{A})^H \in \mathbb{C}^{NP \times NP}. \end{aligned} \quad (3.36)$$

A matriz multimodo descrita na equação (3.36) pode ser vista como uma forma de matriciação multimodo de um tensor de quarta ordem  $\bar{\mathcal{R}} \in \mathbb{C}^{N \times P \times N \times P}$  que mescla os modos-1 e 2 nas linhas e os modos-3 e 4 nas colunas e pode ser escrita em notação de produto modo- $n$  como

$$\bar{\mathcal{R}} = \mathcal{R}_s \times_1 \mathbf{A} \times_2 \mathbf{W} \times_3 \mathbf{A}^* \times_4 \mathbf{W}^*, \quad (3.37)$$

em que  $\mathcal{R}_s \in \mathbb{C}^{M \times M \times M \times M}$  é o tensor de covariância das sequências transmitidas e está relacionado com  $\mathbf{R}_s$  por  $\mathcal{R}_s(:, :, m, m) = \text{diag}(\mathbf{R}_s(:, m))$  para  $m =$

$1, \dots, M$ . O modelo descrito na equação (3.37) é conhecido na literatura como decomposição Tucker4 simétrica [9] do tensor  $\bar{\mathcal{R}}$  em que seu tensor núcleo  $\mathcal{R}_s$  está associado com a matriz de covariância das fontes  $\mathbf{R}_s$ . Nesse caso, as seguintes correspondências são obtidas:

$$\begin{aligned} (I_1, I_2, I_3, I_4, R_1, R_2, R_3, R_4) &\leftrightarrow (N, P, N, P, M, M, M, M) \\ (\mathbf{A}, \mathbf{B}, \mathcal{G}, \mathbf{C}, \mathbf{D}) &\leftrightarrow (\mathbf{A}, \mathbf{W}, \mathcal{R}_s, \mathbf{A}^*, \mathbf{W}^*). \end{aligned} \quad (3.38)$$

Note que a modelagem realizada utilizando uma decomposição Tucker4 absorve naturalmente no tensor núcleo  $\mathcal{R}_s$  a estrutura da matriz de covariância dos sinais transmitidos e assim como na modelagem PARATUCK2 também possibilita a realização da estimação da matriz de assinaturas espaciais em cenários onde as fontes são correlacionadas, desconsiderando a necessidade de se conhecer a estrutura de covariância das sequências transmitidas assim como um número suficientemente grande de *snapshots*. Um caso especial da decomposição Tucker4 ocorre quando as fontes são perfeitamente descorrelacionadas e com variâncias unitárias, ou seja,  $\mathbf{R}_s = \mathbf{I}_M$ . Nesse caso a decomposição Tucker4 é descrita simplificada como

$$\bar{\mathcal{R}} = \mathcal{I}_{4,M} \times_1 \mathbf{A} \times_2 \mathbf{W} \times_3 \mathbf{A}^* \times_4 \mathbf{W}^*, \quad (3.39)$$

em que  $\mathcal{I}_{4,M}$  representa um tensor diagonal com elementos unitários posicionados na superdiagonal [9]. Nesse caso, a covariância espaço-temporal dos dados recebidos é descrita como uma decomposição PARAFAC de um tensor de quarta ordem.

Assim como nos demais métodos apresentados anteriormente o objetivo do algoritmo consiste em estimar a matriz de assinaturas espaciais. O algoritmo ALS-Tucker4 que será descrito a seguir utiliza as formas matriciadas unimodais referentes a cada modo do tensor de covariância espaço-temporal  $\bar{\mathcal{R}}$  descritas por  $[\bar{\mathcal{R}}]_{(1)} \in \mathbb{C}^{N \times NP^2}$ ,  $[\bar{\mathcal{R}}]_{(2)} \in \mathbb{C}^{P \times PN^2}$ ,  $[\bar{\mathcal{R}}]_{(3)} \in \mathbb{C}^{N \times NP^2}$  e  $[\bar{\mathcal{R}}]_{(4)} \in \mathbb{C}^{P \times PN^2}$  e obtidas da seguinte forma [9]

$$[\bar{\mathcal{R}}]_{(1)} = \mathbf{A}[\mathcal{R}_s]_{(1)} (\mathbf{W}^* \otimes \mathbf{A}^* \otimes \mathbf{W})^T, \quad (3.40)$$

$$[\bar{\mathcal{R}}]_{(2)} = \mathbf{W}[\mathcal{R}_s]_{(2)} (\mathbf{W}^* \otimes \mathbf{A}^* \otimes \mathbf{A})^T, \quad (3.41)$$

$$[\bar{\mathcal{R}}]_{(3)} = \mathbf{A}^*[\mathcal{R}_s]_{(3)} (\mathbf{W}^* \otimes \mathbf{W} \otimes \mathbf{A})^T, \quad (3.42)$$

$$[\bar{\mathcal{R}}]_{(4)} = \mathbf{W}^*[\mathcal{R}_s]_{(4)} (\mathbf{A}^* \otimes \mathbf{W} \otimes \mathbf{A})^T, \quad (3.43)$$

em que  $[\mathcal{R}_s]_{(1)} \in \mathbb{C}^{M \times M^3}$ ,  $[\mathcal{R}_s]_{(2)} \in \mathbb{C}^{M \times M^3}$ ,  $[\mathcal{R}_s]_{(3)} \in \mathbb{C}^{M \times M^3}$  e  $[\mathcal{R}_s]_{(4)} \in \mathbb{C}^{M \times M^3}$  são as formas matriciadas do tensor de covariância das fontes  $\mathcal{R}_s$  nos modos 1, 2, 3 e 4, respectivamente. Da mesma forma que os algoritmos ALS-PARAFAC

e ALS-PARATUCK2 o algoritmo ALS-Tucker4 proposto também utiliza um método iterativo baseado em mínimos quadrados alternados e o processo de estimação pode ser realizado assumindo ou não o conhecimento das matrizes  $\mathbf{A}$ ,  $\mathbf{W}$  e  $\mathbf{R}_s$ . De forma geral o objetivo do algoritmo é minimizar a seguinte função custo:

$$f_{\text{custo}}(\hat{\mathbf{R}}_s, \hat{\mathbf{A}}, \hat{\mathbf{W}}) = \left\| [\bar{\mathbf{R}}]_{(1)} - \hat{\mathbf{A}}[\hat{\mathbf{R}}_s]_{(1)} \left( \hat{\mathbf{W}}^* \otimes \hat{\mathbf{A}}^* \otimes \hat{\mathbf{W}} \right)^T \right\|_F^2. \quad (3.44)$$

Note que na equação (3.44) o tensor de covariância das fontes é assumido como sendo desconhecido. Para realizar sua estimação a propriedade descrita em (3.18) é aplicada na equação (3.34), obtendo a seguinte relação

$$\text{vec}(\bar{\mathbf{R}}_{\text{mm}}) = [(\mathbf{W} \diamond \mathbf{A})^* \otimes (\mathbf{W} \diamond \mathbf{A})] \text{vec}(\mathbf{R}_s) \in \mathbb{C}^{N^2 P^2 \times 1}, \quad (3.45)$$

que será utilizada em um dos passos do algoritmo ALS-Tucker4 que será apresentado de forma detalhada a seguir:

#### Algoritmo ALS-Tucker4

1. Fixe  $i = 0$ ;  
Inicialize aleatoriamente  $\hat{\mathcal{G}}_{(i=0)}$ ,  $\hat{\mathbf{A}}_{(i=0)}$ ,  $\hat{\mathbf{B}}_{(i=0)}$ ,  $\hat{\mathbf{C}}_{(i=0)}$  e  $\hat{\mathbf{D}}_{(i=0)}$ ;
2. A partir de  $\hat{\mathcal{G}}_{(i=0)}$  obtenha suas formas matriciadas  $[\hat{\mathcal{G}}_{(i=0)}]_{(1)}$ ,  $[\hat{\mathcal{G}}_{(i=0)}]_{(2)}$ ,  $[\hat{\mathcal{G}}_{(i=0)}]_{(3)}$  e  $[\hat{\mathcal{G}}_{(i=0)}]_{(4)}$ ;
3.  $i = i + 1$ ;
4. Usando  $[\bar{\mathbf{R}}]_{(1)}$  calcule a estimação LS de  $\hat{\mathbf{A}}_{(i)}$  :  
$$\hat{\mathbf{A}}_{(i)} = [\bar{\mathbf{R}}]_{(1)} \left[ [\hat{\mathcal{G}}_{(i-1)}]_{(1)} \left( \hat{\mathbf{D}}_{(i-1)} \otimes \hat{\mathbf{C}}_{(i-1)} \otimes \hat{\mathbf{B}}_{(i-1)} \right)^T \right]^\dagger ;$$
5. Usando  $[\bar{\mathbf{R}}]_{(2)}$  calcule a estimação LS de  $\hat{\mathbf{B}}_{(i)}$  :  
$$\hat{\mathbf{B}}_{(i)} = [\bar{\mathbf{R}}]_{(2)} \left[ [\hat{\mathcal{G}}_{(i-1)}]_{(2)} \left( \hat{\mathbf{D}}_{(i-1)} \otimes \hat{\mathbf{C}}_{(i-1)} \otimes \hat{\mathbf{A}}_{(i)} \right)^T \right]^\dagger ;$$
6. Usando  $[\bar{\mathbf{R}}]_{(3)}$  calcule a estimação LS de  $\hat{\mathbf{C}}_{(i)}$  :  
$$\hat{\mathbf{C}}_{(i)} = [\bar{\mathbf{R}}]_{(3)} \left[ [\hat{\mathcal{G}}_{(i-1)}]_{(3)} \left( \hat{\mathbf{D}}_{(i-1)} \otimes \hat{\mathbf{B}}_{(i)} \otimes \hat{\mathbf{A}}_{(i)} \right)^T \right]^\dagger ;$$
7. Usando  $[\bar{\mathbf{R}}]_{(4)}$  calcule a estimação LS de  $\hat{\mathbf{D}}_{(i)}$  :  
$$\hat{\mathbf{D}}_{(i)} = [\bar{\mathbf{R}}]_{(4)} \left[ [\hat{\mathcal{G}}_{(i-1)}]_{(4)} \left( \hat{\mathbf{C}}_{(i)} \otimes \hat{\mathbf{B}}_{(i)} \otimes \hat{\mathbf{A}}_{(i)} \right)^T \right]^\dagger ;$$
8. A partir de  $\hat{\mathbf{A}}_{(i)}$ ,  $\hat{\mathbf{B}}_{(i)}$ ,  $\hat{\mathbf{C}}_{(i)}$  e  $\hat{\mathbf{D}}_{(i)}$  calcule:  
$$\hat{\Psi}_{1(i)} = \hat{\mathbf{B}}_{(i)} \diamond \hat{\mathbf{A}}_{(i)}; \quad \hat{\Psi}_{2(i)} = \hat{\mathbf{D}}_{(i)} \diamond \hat{\mathbf{C}}_{(i)};$$
9. A partir da relação (3.45) estime  $\hat{\mathcal{G}}_{(i)}$  seguindo os seguintes passos:  
$$\text{vec}(\hat{\mathbf{R}}_s^{(i)}) = \left[ \hat{\Psi}_{1(i)} \otimes \hat{\Psi}_{2(i)} \right]^\dagger \text{vec}(\bar{\mathbf{R}}_{\text{mm}});$$
  
$$\hat{\mathbf{R}}_s^{(i)} = \text{unvec}_{M \times M} \left( \text{vec}(\hat{\mathbf{R}}_s^{(i)}) \right);$$

**for**  $m = 1 : M$

$$\hat{\mathcal{G}}_{(i)}(:, :, m, m) = \text{vec} \left( \hat{\mathcal{R}}_s^{(i)}(:, m) \right);$$

**end**

10. A partir de  $\hat{\mathcal{G}}_{(i)}$  obtenha suas formas matriciadas  $[\hat{\mathcal{G}}_{(i)}]_{(1)}$ ,  $[\hat{\mathcal{G}}_{(i)}]_{(2)}$ ,  $[\hat{\mathcal{G}}_{(i)}]_{(3)}$  e  $[\hat{\mathcal{G}}_{(i)}]_{(4)}$ ;
11. Repetir os passos 3-10 até a convergência.
12. Após a convergência calcule  $\hat{A}_{(\text{final})} = \frac{\hat{A}_{(i)} + \hat{C}_{(i)}^*}{2}$ .

O algoritmo ALS-Tucker4 aborda o caso geral em que o tensor núcleo e todas as matrizes fatores são desconhecidas. De forma semelhante aos algoritmos ALS-PARAFAC e ALS-PARATUCK2, ou seja, assumindo que a matriz de potências  $W$  é conhecida no receptor é fácil perceber que os passos 5 e 7 são omitidos durante o processo de estimação. Assim como nos demais casos a convergência do algoritmo ALS-Tucker4 é alcançada quando o erro residual entre duas iterações sucessivas for  $|e_{(i)} - e_{(i-1)}| \leq 10^{-6}$  em que:

$$e_{(i)} = \left\| [\bar{\mathcal{R}}]_{(1)} - \hat{A}[\hat{\mathcal{R}}_s]_{(1)} \left( \hat{W}^* \otimes \hat{A}^* \otimes \hat{W} \right)^T \right\|_F^2. \quad (3.46)$$

### 3.4 Condições de Identificabilidade

Como o objetivo dos métodos propostos é realizar a estimação cega dos parâmetros de interesse, a identificabilidade torna-se uma questão importante que deve ser considerada. Nesta seção, serão analisadas as condições de identificabilidade das matrizes de assinaturas espaciais e de covariância das fontes ( $A$  e  $R_s$ ) utilizando os receptores PARAFAC, PARATUCK2 e Tucker4 descritos anteriormente.

**Receptor PARAFAC:** Nesse caso, a identificabilidade no sentido dos mínimos quadrados está ligada a recuperação das matrizes  $A$  e  $R_s$  a partir das equações (3.7), (3.8) e (3.12), respectivamente. De acordo com as seguintes equações

$$[\bar{\mathcal{R}}]_{(1)} = A (\bar{W} R_s \diamond A^*)^T \quad \text{e} \quad [\bar{\mathcal{R}}]_{(2)} = A^* (\bar{W} R_s \diamond A)^T,$$

é possível perceber que uma solução de mínimos quadrados para  $A$  exige que  $(\bar{W} R_s \diamond A^*)^T$  e  $(\bar{W} R_s \diamond A)^T$  sejam *rank* linha completo e invertíveis à direita, de forma que a seguinte condição necessária deve ser satisfeita:

$$PN \geq M. \quad (3.47)$$

Com relação a recuperação da matriz  $R_s$  a análise dos critérios de identificabilidade é realizada a partir de

$$\text{vec}([\bar{\mathcal{R}}]_{(3)}) = (\mathbf{A}^* \diamond \mathbf{A} \diamond \bar{\mathbf{W}}) \text{vecd}(R_s).$$

Nesse caso, percebe-se a exigência de que  $(\mathbf{A}^* \diamond \mathbf{A} \diamond \bar{\mathbf{W}})$  seja *rank* coluna completo e invertível à esquerda, para que estas condições sejam satisfeitas é necessário garantir que

$$PN^2 \geq M. \quad (3.48)$$

**Receptor PARATUCK2:** A análise da identificabilidade para este caso é realizada de forma similar a ideia apresentada para o receptor PARAFAC. Entretanto, tais condições são deduzidas a partir das equações (3.23), (3.26) e (3.27) e de acordo com

$$[\bar{\mathcal{R}}]_{(2)} = (\mathbf{I}_P \otimes \mathbf{A}) \mathbf{F}_2 \mathbf{A}^H \quad \text{e} \quad [\bar{\mathcal{R}}]_{(3)} = (\mathbf{I}_P \otimes \mathbf{A}^*) \mathbf{F}_3 \mathbf{A}^T,$$

percebe-se que para a obtenção de  $\mathbf{A}$  os termos  $(\mathbf{I}_P \otimes \mathbf{A}) \mathbf{F}_2$  e  $(\mathbf{I}_P \otimes \mathbf{A}^*) \mathbf{F}_3$  devem ser *rank* coluna completo e invertíveis à esquerda, conduzindo a seguinte desigualdade:

$$PN \geq M, \quad (3.49)$$

que é a mesma condição necessária obtida anteriormente para o receptor PARAFAC. Com relação a  $R_s$  temos que

$$\text{vec}([\bar{\mathcal{R}}]_{(1)}) = [(\mathbf{W}^H \diamond \mathbf{W}^T)^T \diamond (\mathbf{A}^* \otimes \mathbf{A})] \text{vec}(R_s).$$

Nesse caso, uma solução de mínimos quadrados exige que  $[(\mathbf{W}^H \diamond \mathbf{W}^T)^T \diamond (\mathbf{A}^* \otimes \mathbf{A})]$  seja *rank* coluna completo e invertível à esquerda. tais características podem ser obtidas quando

$$PN^2 \geq M^2. \quad (3.50)$$

**Receptor Tucker4:** Por último, as condições de identificabilidade do receptor Tucker4 são deduzidas a partir das equações (3.40), (3.42) e (3.45). Note que de acordo com as seguintes expressões

$$[\bar{\mathcal{R}}]_{(1)} = \mathbf{A}[\mathcal{R}_s]_{(1)} (\mathbf{W}^* \otimes \mathbf{A}^* \otimes \mathbf{W})^T \quad \text{e} \quad [\bar{\mathcal{R}}]_{(3)} = \mathbf{A}^*[\mathcal{R}_s]_{(3)} (\mathbf{W}^* \otimes \mathbf{W} \otimes \mathbf{A})^T,$$

uma solução para  $\mathbf{A}$  é obtida quando  $[\mathcal{R}_s]_{(1)} (\mathbf{W}^* \otimes \mathbf{A}^* \otimes \mathbf{W})^T$  e  $[\mathcal{R}_s]_{(3)} (\mathbf{W}^* \otimes \mathbf{W} \otimes \mathbf{A})^T$  forem *rank* linha completo e invertíveis à direita, o

que nos leva a obtenção da seguinte condição:

$$P^2N \geq M. \quad (3.51)$$

Para a obtenção da matriz  $R_s$  a seguinte equação é analisada

$$\text{vec}(\bar{R}_{\text{mm}}) = [(\mathbf{W} \diamond \mathbf{A})^* \otimes (\mathbf{W} \diamond \mathbf{A})] \text{vec}(\mathbf{R}_s),$$

percebe-se a exigência de que  $[(\mathbf{W} \diamond \mathbf{A})^* \otimes (\mathbf{W} \diamond \mathbf{A})]$  seja *rank* coluna completo e invertível à esquerda, nesse caso temos que

$$PN \geq M. \quad (3.52)$$

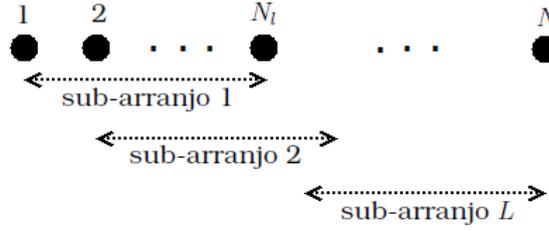
No decorrer deste trabalho é assumido **DoA** distintas para cada fonte considerada. Além disso, assume-se também que a matriz de potências  $W$  é composta por valores reais positivos e aleatórios que seguem uma distribuição uniforme. Tais considerações garantem, portanto, que as matrizes  $A$  e  $W$  possuem *rank* completo. De acordo com os critérios de identificabilidade obtidos anteriormente para os diferentes receptores, conclui-se que a existência da solução de mínimos quadrados é assegurada quando  $P > M$ .

### 3.5 Estimação da DoA a partir de $\hat{A}_{(\text{final})}$

Note que a estrutura da matriz de assinaturas espaciais  $A$  foi considerada desconhecida no decorrer de toda a descrição dos algoritmos ALS-PARAFAC, ALS-PARATUCK2 e ALS-Tucker4. Esta característica é de fundamental importância nos casos em que o arranjo de sensores possui uma geometria arbitrária de forma que uma estimação  $\hat{A}_{(\text{final})}$  pode ser facilmente obtida utilizando os algoritmos propostos. Para as configurações *2L-Shape* e ULA descritas no Capítulo 2, estimações para as matrizes de assinaturas espaciais  $\hat{A}_x$ ,  $\hat{A}_y$  e  $\hat{A}_z$  são obtidas a partir de  $\hat{A}_{(\text{final})}$  utilizando as seguintes relações:

$$\hat{A}_{(\text{final})} = \begin{cases} \begin{bmatrix} \hat{A}_x \\ \hat{A}_y \\ \hat{A}_z \end{bmatrix}, \text{ para o arranjo } 2L\text{-Shape} \\ \hat{A}_z, \text{ para ULA.} \end{cases} \quad (3.53)$$

Naturalmente as estimações para os ângulos de elevação e azimute referentes a  $m$ -ésima fonte são obtidos considerando a média entre as estimações provenientes de cada sensor independentemente. Dessa forma,



**Figura 3.3:** *Spatial Smoothing* para um Arranjo Arbitrário.

a estimação do ângulo de elevação  $\hat{\alpha}_m$  é obtida a partir de

$$\hat{\alpha}_m = \frac{1}{N_z - 1} \sum_{n_z=2}^{N_z} \arccos \left[ \frac{\arg(\hat{A}_{z(n_z,m)})}{2\pi(n_z - 1)d_z/\lambda} \right]. \quad (3.54)$$

Utilizando  $\hat{\alpha}_m$  duas estimações independentes  $\hat{\beta}_m^{(x)}$  e  $\hat{\beta}_m^{(y)}$  para o ângulo de azimute da  $m$ -ésima fonte são obtidas utilizando as seguintes equações:

$$\hat{\beta}_m^{(x)} = \frac{1}{N_x - 1} \sum_{n_x=2}^{N_x} \arccos \left[ \frac{\arg(\hat{A}_{x(n_x,m)})}{(2\pi(n_x - 1)d_x \sin \hat{\alpha}_m)/\lambda} \right], \quad (3.55)$$

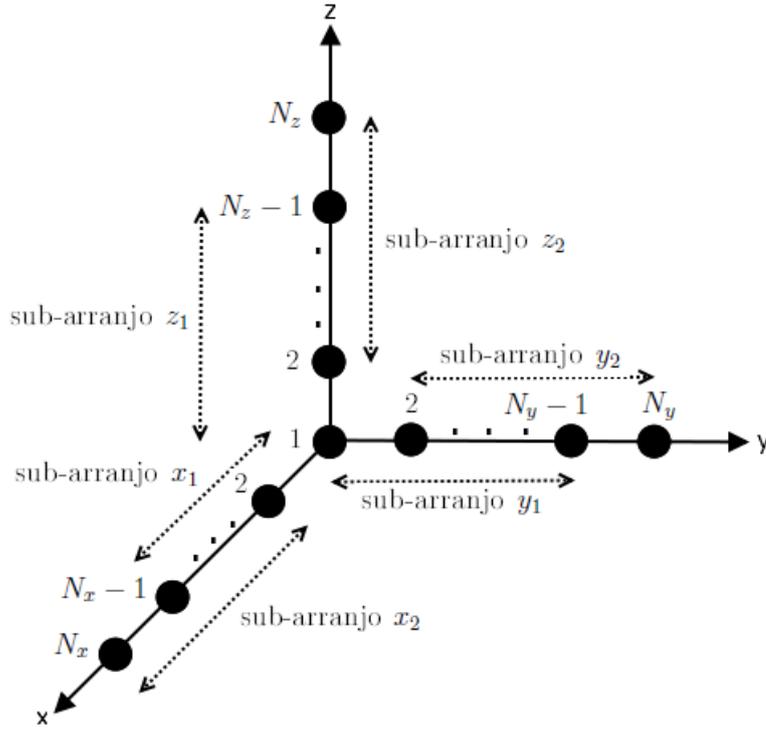
$$\hat{\beta}_m^{(y)} = \frac{1}{N_y - 1} \sum_{n_y=2}^{N_y} \arcsin \left[ \frac{\arg(\hat{A}_{y(n_y,m)})}{(2\pi(n_y - 1)d_y \sin \hat{\alpha}_m)/\lambda} \right], \quad (3.56)$$

por último, de forma semelhante ao algoritmo ESPRIT para o arranjo *2L-Shape* uma estimativa para  $\hat{\beta}_m$  é obtida utilizando as relações descritas em (2.31). Em virtude da possibilidade de utilização dos algoritmos ALS-PARAFAC, ALS-PARATUCK2 e ALS-Tucker4 para arranjos com geometria arbitrária, serão apresentadas a seguir, outras propostas de receptores que possibilitam realizar a estimação da DoA dos sinais incidentes utilizando estes métodos.

### 3.6 Arranjo 2L-Shape com Spatial Smoothing

O método que será descrito nesta seção é baseado na utilização da propriedade de invariância ao deslocamento. Esta propriedade possibilita realizar uma abordagem ligeiramente modificada em relação aos métodos tensoriais descritos anteriormente. De forma geral, vamos supor que um arranjo arbitrário seja dividido em  $L$  sub-arranjos idênticos compostos por  $N_l$  sensores cada, como ilustra a Figura 3.3. A partir destas considerações, a matriz dos dados recebidos no  $l$ -ésimo sub-arranjo é dada por [4]

$$\mathbf{X}_l = \underbrace{\mathbf{J}_l \mathbf{A}}_{\mathbf{A}' \Phi_l} \mathbf{S} + \underbrace{\mathbf{J}_l \mathbf{V}}_{\mathbf{V}_l} \rightarrow \mathbf{X}_l = \mathbf{A}' \Phi_l \mathbf{S} + \mathbf{V}_l \in \mathbb{C}^{N_l \times T},$$



**Figura 3.4:** Divisão do Arranjo 2L-Shape em Sub-Arranjos.

$$\mathbf{X}_l = \mathbf{A}' D_l(\Phi) \mathbf{S} + \mathbf{V}_l, \quad l = 1, \dots, L. \quad (3.57)$$

em que  $\mathbf{J}_l$  é uma matriz de seleção de ordem  $N_l \times N$  utilizada para selecionar os sensores que compõem o  $l$ -ésimo sub-arranjo,  $\mathbf{A}' \in \mathbb{C}^{N_l \times M}$  é a matriz de assinaturas espaciais do sub-arranjo de referência e  $\mathbf{V}_l \in \mathbb{C}^{N_l \times T}$  é a matriz de ruídos do  $l$ -ésimo sub-arranjo. Note que  $\Phi_l \in \mathbb{C}^{M \times M}$  é uma matriz diagonal que contém a defasagem sofrida pelo  $l$ -ésimo sub-arranjo em relação ao sub-arranjo de referência e sua diagonal compõe a  $l$ -ésima linha de  $\Phi \in \mathbb{C}^{L \times M}$ . Desconsiderando o ruído aditivo na equação (3.57), os dados recebidos no  $l$ -ésimo sub-arranjo  $\bar{\mathbf{X}}_l = \mathbf{X}_l - \mathbf{V}_l$  podem ser concatenados para  $l = 1, \dots, L$  obtendo uma única matriz  $\mathbf{X} \in \mathbb{C}^{N_L \times T}$  descrita por

$$\underbrace{\begin{bmatrix} \bar{\mathbf{X}}_1 \\ \bar{\mathbf{X}}_2 \\ \vdots \\ \bar{\mathbf{X}}_L \end{bmatrix}}_{\mathbf{X}} = \begin{bmatrix} \mathbf{A}' D_1(\Phi) \\ \mathbf{A}' D_2(\Phi) \\ \vdots \\ \mathbf{A}' D_L(\Phi) \end{bmatrix} \mathbf{S} \rightarrow \mathbf{X} = (\Phi \diamond \mathbf{A}') \mathbf{S} \in \mathbb{C}^{N_L \times T}. \quad (3.58)$$

A ideia principal desta seção é realizar a modelagem do receptor para um arranjo 2L-Shape utilizando a técnica *Spatial Smoothing*. Para uma descrição geral da modelagem vamos assumir que os arranjos lineares posicionados nos eixos  $x$ ,  $y$  e  $z$  são divididos em  $L$  sub-arranjos idênticos com  $N_l$  sensores cada

(a Figura 3.4 ilustra o caso em que  $L = 2$ ) e considerando por simplicidade de notação que  $L$  e  $N_l$  são independentes para cada eixo. Dessa forma, os dados recebidos no  $l$ -ésimo sub-arranjo dos eixos  $x$ ,  $y$  e  $z$  no  $p$ -ésimo sub-bloco são descritos respectivamente por

$$\begin{aligned}\mathbf{X}_{(l,p)} &= \mathbf{A}'_x D_l(\Phi_x) D_p(\mathbf{W}) \mathbf{S}, \\ \mathbf{Y}_{(l,p)} &= \mathbf{A}'_y D_l(\Phi_y) D_p(\mathbf{W}) \mathbf{S}, \\ \mathbf{Z}_{(l,p)} &= \mathbf{A}'_z D_l(\Phi_z) D_p(\mathbf{W}) \mathbf{S},\end{aligned}\tag{3.59}$$

em que  $\mathbf{A}'_x$ ,  $\mathbf{A}'_y$  e  $\mathbf{A}'_z \in \mathbb{C}^{N_l \times M}$  são as matrizes de assinaturas espaciais dos sub-arranjos de referência e  $\Phi_x$ ,  $\Phi_y$  e  $\Phi_z \in \mathbb{C}^{L \times M}$  são as matrizes de defasagem, ambas relacionadas aos eixos  $x$ ,  $y$  e  $z$ , respectivamente. A partir da equação (3.59) os dados recebidos nos  $L$  sub-arranjos de cada eixo são obtidos de forma semelhante a equação (3.58), de forma que para o eixo- $x$  obtemos:

$$\underbrace{\begin{bmatrix} \mathbf{X}_{(1,p)} \\ \mathbf{X}_{(2,p)} \\ \vdots \\ \mathbf{X}_{(L,p)} \end{bmatrix}}_{\mathbf{X}_p} = \begin{bmatrix} \mathbf{A}'_x D_1(\Phi_x) \\ \mathbf{A}'_x D_2(\Phi_x) \\ \vdots \\ \mathbf{A}'_x D_L(\Phi_x) \end{bmatrix} D_p(\mathbf{W}) \mathbf{S} \rightarrow \mathbf{X}_p = (\Phi_x \diamond \mathbf{A}'_x) D_p(\mathbf{W}) \mathbf{S},\tag{3.60}$$

para o eixo- $y$ :

$$\underbrace{\begin{bmatrix} \mathbf{Y}_{(1,p)} \\ \mathbf{Y}_{(2,p)} \\ \vdots \\ \mathbf{Y}_{(L,p)} \end{bmatrix}}_{\mathbf{Y}_p} = \begin{bmatrix} \mathbf{A}'_y D_1(\Phi_y) \\ \mathbf{A}'_y D_2(\Phi_y) \\ \vdots \\ \mathbf{A}'_y D_L(\Phi_y) \end{bmatrix} D_p(\mathbf{W}) \mathbf{S} \rightarrow \mathbf{Y}_p = (\Phi_y \diamond \mathbf{A}'_y) D_p(\mathbf{W}) \mathbf{S},\tag{3.61}$$

e por último para o eixo- $z$ :

$$\underbrace{\begin{bmatrix} \mathbf{Z}_{(1,p)} \\ \mathbf{Z}_{(2,p)} \\ \vdots \\ \mathbf{Z}_{(L,p)} \end{bmatrix}}_{\mathbf{Z}_p} = \begin{bmatrix} \mathbf{A}'_z D_1(\Phi_z) \\ \mathbf{A}'_z D_2(\Phi_z) \\ \vdots \\ \mathbf{A}'_z D_L(\Phi_z) \end{bmatrix} D_p(\mathbf{W}) \mathbf{S} \rightarrow \mathbf{Z}_p = (\Phi_z \diamond \mathbf{A}'_z) D_p(\mathbf{W}) \mathbf{S}.\tag{3.62}$$

**Exemplo:** Para exemplificar a obtenção das matrizes  $\Phi_x$ ,  $\Phi_y$  e  $\Phi_z$  vamos considerar o arranjo ilustrado na Figura 3.4. De forma geral, para um

arranjo linear uniforme quando o deslocamento realizado em cada sub-arranjo em relação ao anterior for  $d_x, d_y$  e  $d_z$  as defasagens sofridas pelo  $l$ -ésimo sub-arranjo são descritas por:

$$\begin{aligned}\Phi_x(l, :) &= \left[ e^{-j\frac{2\pi}{\lambda}(l-1)d_x \sin \alpha_1 \cos \beta_1}, \dots, e^{-j\frac{2\pi}{\lambda}(l-1)d_x \sin \alpha_M \cos \beta_M} \right]; \\ \Phi_y(l, :) &= \left[ e^{-j\frac{2\pi}{\lambda}(l-1)d_y \sin \alpha_1 \sin \beta_1}, \dots, e^{-j\frac{2\pi}{\lambda}(l-1)d_y \sin \alpha_M \sin \beta_M} \right]; \\ \Phi_z(l, :) &= \left[ e^{-j\frac{2\pi}{\lambda}(l-1)d_z \cos \alpha_1}, \dots, e^{-j\frac{2\pi}{\lambda}(l-1)d_z \cos \alpha_M} \right].\end{aligned}\quad (3.63)$$

Note que os termos  $(\Phi_x \diamond \mathbf{A}'_x)$ ,  $(\Phi_y \diamond \mathbf{A}'_y)$  e  $(\Phi_z \diamond \mathbf{A}'_z)$  presentes nas equações (3.60), (3.61) e (3.62) representam as formas matriciadas dos seguintes tensores descritos pela decomposição PARAFAC [9]:

$$\begin{aligned}\mathcal{F}_1 &= \mathcal{I}_{3,M} \times_1 \mathbf{I} \times_2 \mathbf{A}'_x \times_3 \Phi_x \rightarrow [\mathcal{F}_1]_{(1)}^T = (\Phi_x \diamond \mathbf{A}'_x), \\ \mathcal{F}_2 &= \mathcal{I}_{3,M} \times_1 \mathbf{I} \times_2 \mathbf{A}'_y \times_3 \Phi_y \rightarrow [\mathcal{F}_2]_{(1)}^T = (\Phi_y \diamond \mathbf{A}'_y), \\ \mathcal{F}_3 &= \mathcal{I}_{3,M} \times_1 \mathbf{I} \times_2 \mathbf{A}'_z \times_3 \Phi_z \rightarrow [\mathcal{F}_3]_{(1)}^T = (\Phi_z \diamond \mathbf{A}'_z),\end{aligned}\quad (3.64)$$

Utilizando as relações descritas anteriormente, os dados recebidos nos  $L$  sub-arranjos de cada eixo podem ser concatenados afim de se obter uma única matriz referente ao  $p$ -ésimo sub-bloco, como descrito a seguir

$$\underbrace{\begin{bmatrix} \mathbf{X}_p \\ \mathbf{Y}_p \\ \mathbf{Z}_p \end{bmatrix}}_{\mathbf{X}_{(p)}} = \begin{bmatrix} [\mathcal{F}_1]_{(1)}^T \\ [\mathcal{F}_2]_{(1)}^T \\ [\mathcal{F}_3]_{(1)}^T \end{bmatrix} D_p(\mathbf{W})\mathbf{S}, \quad p = 1, \dots, P. \quad (3.65)$$

A partir da equação (3.65) calculamos a matriz de covariância dos dados recebidos no  $p$ -ésimo sub-bloco

$$\bar{\mathbf{R}}_{(p)} \triangleq \frac{1}{T_s} \mathbf{X}_{(p)} \mathbf{X}_{(p)}^H \rightarrow \bar{\mathbf{R}}_{(p)} = \begin{bmatrix} [\mathcal{F}_1]_{(1)}^T \\ [\mathcal{F}_2]_{(1)}^T \\ [\mathcal{F}_3]_{(1)}^T \end{bmatrix} D_p(\mathbf{W})\mathbf{R}_s D_p(\mathbf{W}^*) \begin{bmatrix} [\mathcal{F}_1]_{(1)}^T \\ [\mathcal{F}_2]_{(1)}^T \\ [\mathcal{F}_3]_{(1)}^T \end{bmatrix}^H. \quad (3.66)$$

Note que a equação (3.66) possui a estrutura de uma decomposição PARATUCK2 e representa um dos modelos de recepção do arranjo 2L-Shape com Spatial Smoothing. Entretanto, a partir da equação (3.65) os mesmos procedimentos realizados na descrição dos receptores PARAFAC e Tucker4 podem ser repetidos, obtendo dessa forma as representações do receptor para o arranjo 2L-Shape com Spatial Smoothing em termos das decomposições tensoriais PARAFAC e Tucker4 descritos a seguir:

$$\bar{\mathbf{R}}_{(p)} = \begin{bmatrix} [\mathcal{F}_1]_{(1)}^T \\ [\mathcal{F}_2]_{(1)}^T \\ [\mathcal{F}_3]_{(1)}^T \end{bmatrix} D_p(\Omega) \begin{bmatrix} [\mathcal{F}_1]_{(1)}^T \\ [\mathcal{F}_2]_{(1)}^T \\ [\mathcal{F}_3]_{(1)}^T \end{bmatrix}^H \rightarrow \text{PARAFAC} \quad (3.67)$$

e

$$\bar{\mathbf{R}} = \mathcal{R}_s \times_1 \begin{bmatrix} [\mathcal{F}_1]_{(1)}^T \\ [\mathcal{F}_2]_{(1)}^T \\ [\mathcal{F}_3]_{(1)}^T \end{bmatrix} \times_2 \mathbf{W} \times_3 \begin{bmatrix} [\mathcal{F}_1]_{(1)}^T \\ [\mathcal{F}_2]_{(1)}^T \\ [\mathcal{F}_3]_{(1)}^T \end{bmatrix}^* \times_4 \mathbf{W}^* \rightarrow \text{Tucker4}. \quad (3.68)$$

Nesse caso a forma matriciada multimodo  $\bar{\mathbf{R}}_{\text{mm}}$  do tensor  $\bar{\mathbf{R}}$  é dada por

$$\bar{\mathbf{R}}_{\text{mm}} = \left( \mathbf{W} \otimes \begin{bmatrix} [\mathcal{F}_1]_{(1)}^T \\ [\mathcal{F}_2]_{(1)}^T \\ [\mathcal{F}_3]_{(1)}^T \end{bmatrix} \right) [\mathcal{R}_s]_{(1,2);(3,4)} \left( \mathbf{W} \otimes \begin{bmatrix} [\mathcal{F}_1]_{(1)}^T \\ [\mathcal{F}_2]_{(1)}^T \\ [\mathcal{F}_3]_{(1)}^T \end{bmatrix} \right)^H, \quad (3.69)$$

de forma que a matriz de assinaturas espaciais  $\mathbf{A}$  para esta configuração possui a seguinte estrutura:

$$\mathbf{A} = \begin{bmatrix} [\mathcal{F}_1]_{(1)}^T \\ [\mathcal{F}_2]_{(1)}^T \\ [\mathcal{F}_3]_{(1)}^T \end{bmatrix}. \quad (3.70)$$

Como extensões naturais do arranjo *2L-Shape*, a seguir serão apresentadas as modelagens tensoriais para o arranjo planar considerando dois casos distintos. No primeiro caso será realizada uma modelagem para o arranjo ilustrado na Figura 2.3, enquanto no segundo a técnica *Spatial Smoothing* será utilizada para este mesmo arranjo. Como será visto a seguir, os dados recebidos no arranjo de sensores são modelados como decomposições PARAFAC, PARATUCK2 E Tucker4 de forma que a matriz  $\mathbf{A}$ , diferente em cada situação, pode ser estimada utilizando os algoritmos propostos anteriormente.

### 3.7 Arranjo Planar sem *Spatial Smoothing*

Para o arranjo planar ilustrado na Figura 2.3 o sinal recebido no  $(n_x, n_z)$ -ésimo sensor proveniente da  $m$ -ésima fonte é descrito pela equação (2.11). A partir dessa estrutura vamos considerar que o  $m$ -ésimo sinal recebido no  $p$ -ésimo sub-bloco possua uma potência  $w_{p,m}$  de forma que a equação (2.11) seja reescrita da seguinte forma desprezando a influência do ruído

$$x_{n_x, n_z, p}(t) = s_m(t) \cdot w_{p,m} \cdot \underbrace{e^{-j \frac{2\pi}{\lambda} (n_x - 1) d_x \sin \alpha_m \cos \beta_m}}_{a_{n_x, m}^{(x)}} \cdot \underbrace{e^{-j \frac{2\pi}{\lambda} (n_z - 1) d_z \cos \alpha_m}}_{a_{n_z, m}^{(z)}},$$

$$x_{n_x, n_z, p}(t) = s_m(t) \cdot w_{p,m} \cdot a_{n_x, m}^{(x)} \cdot a_{n_z, m}^{(z)}, \quad (3.71)$$

em que o termo  $a_{n_x, m}^{(x)} \cdot a_{n_z, m}^{(z)}$  representa a defasagem sofrida pelo  $m$ -ésimo sinal recebido no  $(n_x, n_z)$ -ésimo sensor em relação ao sensor de referência. A partir da equação (3.71) a matriz dos dados recebidos  $\mathbf{X} \in \mathbb{C}^{N_x \times N_z}$  referente ao  $t$ -ésimo *snapshot* do  $p$ -ésimo sub-bloco é descrita por

$$\mathbf{X} = \begin{bmatrix} x_{1,1,p}(t) & \cdots & x_{1,N_z,p}(t) \\ \vdots & \ddots & \vdots \\ x_{N_x,1,p}(t) & \cdots & x_{N_x,N_z,p}(t) \end{bmatrix}, \quad n_x = 1, \dots, N_x \quad \text{e} \quad n_z = 1, \dots, N_z. \quad (3.72)$$

em que

$$\mathbf{X} = \mathbf{a}_x(\alpha_m, \beta_m) \mathbf{a}_z^T(\alpha_m) s_m(t) w_{p,m} \in \mathbb{C}^{N_x \times N_z}. \quad (3.73)$$

Para  $M$  sinais incidentes simultaneamente, os dados recebidos são obtidos realizando o somatório de todas as contribuições, ou seja, para  $m = 1, \dots, M$  obtemos

$$\begin{aligned} \mathbf{X} &= \sum_{m=1}^M \mathbf{a}_x(\alpha_m, \beta_m) \mathbf{a}_z^T(\alpha_m) s_m(t) w_{p,m} & (3.74) \\ &= \begin{bmatrix} \mathbf{a}_x(\alpha_1, \beta_1) & \cdots & \mathbf{a}_x(\alpha_M, \beta_M) \end{bmatrix} \begin{bmatrix} s_1(t) \\ \vdots \\ s_M(t) \end{bmatrix} \begin{bmatrix} w_{p,1} & \cdots & w_{p,M} \end{bmatrix} \begin{bmatrix} \mathbf{a}_z^T(\alpha_1) \\ \vdots \\ \mathbf{a}_z^T(\alpha_M) \end{bmatrix}, \end{aligned}$$

de forma sumarizada a equação (3.74) pode ser reescrita da seguinte forma

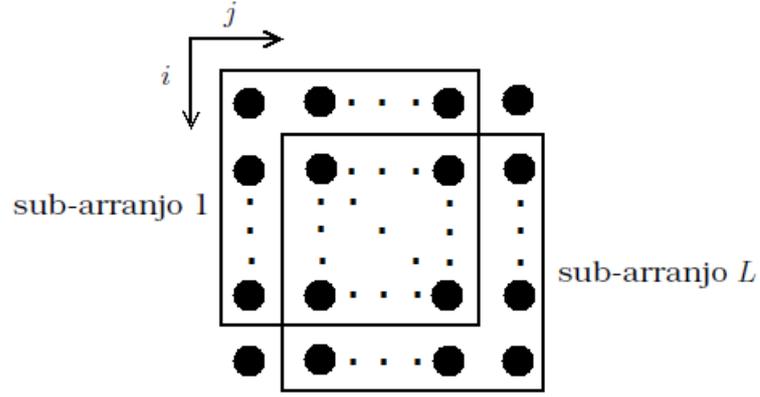
$$\mathbf{X}_{..pt} = \mathbf{A}_x D_p(\mathbf{W}) D_t(\mathbf{S}^T) \mathbf{A}_z^T. \quad (3.75)$$

Note que de acordo com a equação (2.51) os dados recebidos no arranjo planar são descritos pela decomposição PARAFAC do tensor de quarta ordem  $\mathcal{X} \in \mathbb{C}^{N_x \times N_z \times P \times T_s}$  em que a equação (3.75) representa os *slices* de  $\mathcal{X}$  obtidos fixando as dimensões  $N_x$  e  $N_z$  e variando  $t = 1, \dots, T_s$  e  $p = 1, \dots, P$ . Para obter estruturas compatíveis com os algoritmos propostos, vamos inicialmente reescrever a equação (3.75) utilizando a propriedade do produto de Khatri-Rao descrita em (3.11) obtendo a seguinte equação

$$\text{vec}(\mathbf{X}_{..pt}) = (\mathbf{A}_z \diamond \mathbf{A}_x) D_p(\mathbf{W}) \text{vecd}(D_t(\mathbf{S}^T)), \quad (3.76)$$

em seguida concatenando os dados recebidos no decorrer dos  $T_s$  *snapshots* que compõem o  $p$ -ésimo sub-bloco temos

$$\begin{aligned} \left[ \text{vec}(\mathbf{X}_{..p1}) \quad \cdots \quad \text{vec}(\mathbf{X}_{..pT_s}) \right] &= (\mathbf{A}_z \diamond \mathbf{A}_x) D_p(\mathbf{W}) \left[ \text{vecd}(D_1(\mathbf{S}^T)) \quad \cdots \quad \text{vecd}(D_{T_s}(\mathbf{S}^T)) \right], \\ \mathbf{Y}_{(p)} &= [\mathcal{H}_1]_{(1)}^T D_p(\mathbf{W}) \mathbf{S}, \quad p = 1, \dots, P. & (3.77) \end{aligned}$$



**Figura 3.5:** *Spatial Smoothing* para o Arranjo Planar.

em que  $\mathbf{Y}_{(p)} \in \mathbb{C}^{N_x N_z \times T_s}$  é uma forma equivalente da matriz dos dados recebidos no arranjo planar no  $p$ -ésimo sub-bloco e  $[\mathcal{H}_1]_{(1)}^T = (\mathbf{A}_z \diamond \mathbf{A}_x)$  representa a forma matriciada no modo-1 do seguinte tensor PARAFAC [9]:

$$\mathcal{H}_1 = \mathcal{I}_{3,M} \times_1 \mathbf{I} \times_2 \mathbf{A}_x \times_3 \mathbf{A}_z \rightarrow [\mathcal{H}_1]_{(1)}^T = (\mathbf{A}_z \diamond \mathbf{A}_x). \quad (3.78)$$

A partir da equação (3.77) obtemos a matriz de covariância espacial dos dados recebidos no  $p$ -ésimo sub-bloco

$$\bar{\mathbf{R}}_{(p)} \triangleq \frac{1}{T_s} \mathbf{Y}_{(p)} \mathbf{Y}_{(p)}^H \rightarrow \bar{\mathbf{R}}_{(p)} = \left[ [\mathcal{H}_1]_{(1)}^T \right] D_p(\mathbf{W}) \mathbf{R}_s D_p(\mathbf{W}^*) \left[ [\mathcal{H}_1]_{(1)}^T \right]^H. \quad (3.79)$$

A equação (3.79) descreve a representação do receptor para o arranjo planar em termos da decomposição PARATUCK2 do tensor  $\bar{\mathcal{R}} \in \mathbb{C}^{N_x N_z \times N_x N_z \times P}$ . De forma análoga as Seções 3.1 e 3.2 as representações do modelo de recepção em termos das decomposições PARAFAC e Tucker4 são obtidas:

$$\bar{\mathbf{R}}_{(p)} = \left[ [\mathcal{H}_1]_{(1)}^T \right] D_p(\Omega) \left[ [\mathcal{H}_1]_{(1)}^T \right]^H \rightarrow \text{PARAFAC} \quad (3.80)$$

e

$$\bar{\mathcal{R}} = \mathcal{R}_s \times_1 \left[ [\mathcal{H}_1]_{(1)}^T \right] \times_2 \mathbf{W} \times_3 \left[ [\mathcal{H}_1]_{(1)}^T \right]^* \times_4 \mathbf{W}^* \rightarrow \text{Tucker4}. \quad (3.81)$$

Para esse caso as matrizes  $\bar{\mathbf{R}}_{\text{mm}}$  e  $\mathbf{A}$  são descritas da seguinte forma respectivamente:

$$\bar{\mathbf{R}}_{\text{mm}} = (\mathbf{W} \otimes [\mathcal{H}_1]_{(1)}^T) [\mathcal{R}_s]_{(1,2);(3,4)} (\mathbf{W} \otimes [\mathcal{H}_1]_{(1)}^T)^H \quad \text{e} \quad \mathbf{A} = [\mathcal{H}_1]_{(1)}^T. \quad (3.82)$$

### 3.8 Arranjo Planar com *Spatial Smoothing*

A segunda abordagem referente ao arranjo planar é baseada na utilização da técnica *Spatial Smoothing*. De forma semelhante a modelagem realizada na

Seção 3.6, vamos considerar que o arranjo planar ilustrado na Figura 2.3 seja dividido em  $L$  sub-arranjos idênticos de acordo com a Figura 3.5.

Por se tratar de um arranjo em duas dimensões existem duas matrizes relacionadas com as defasagens sofridas pelos sub-arranjos de forma que a  $i$ -ésima linha de  $\Phi_x \in \mathbb{C}^{I \times M}$  possui a defasagem sofrida em relação ao eixo- $x$  e a  $j$ -ésima linha de  $\Phi_z \in \mathbb{C}^{J \times M}$  possui a defasagem sofrida em relação ao eixo- $z$ , de forma que o par  $(i, j)$  está relacionado com o  $l$ -ésimo sub-arranjo. A partir dessas considerações a matriz dos dados recebidos no  $l$ -ésimo sub-arranjo é descrita da seguinte forma

$$\mathbf{X}_{..pt} = (\mathbf{J}_i \mathbf{A}_x) D_p(\mathbf{W}) D_t(\mathbf{S}^T) (\mathbf{J}_j \mathbf{A}_z)^T, \quad (3.83)$$

em que  $\mathbf{J}_i$  e  $\mathbf{J}_j$  são matrizes de seleção utilizadas para selecionar os sensores relacionados aos eixos  $x$  e  $z$  que compõem o  $l$ -ésimo sub-arranjo, respectivamente. Os  $L$  sub-arranjos considerados possuem invariância dual [28], dessa forma podemos reescrever a equação (3.83) de forma mais geral, ou seja,

$$\begin{aligned} \mathbf{X}_{(i,p,t,j)} &= (\mathbf{A}'_x D_i(\Phi_x)) D_p(\mathbf{W}) D_t(\mathbf{S}^T) (\mathbf{A}'_z D_j(\Phi_z))^T \quad \therefore \\ \mathbf{X}_{(i,p,t,j)} &= \mathbf{A}'_x D_i(\Phi_x) D_p(\mathbf{W}) D_t(\mathbf{S}^T) D_j(\Phi_z) \mathbf{A}'_z{}^T. \end{aligned} \quad (3.84)$$

Note que os dados recebidos no  $l$ -ésimo sub-arranjo são modelados pela decomposição PARAFAC do tensor  $\mathcal{X}$  de ordem 6. Entretanto, tensores que possuem estruturas compatíveis com os algoritmos propostos podem ser obtidos a partir da equação (3.84). Para descrevê-los devemos inicialmente agrupar os modos 1 e 6 do tensor  $\mathcal{X}$  nas linhas para  $i = 1, \dots, I$  realizando a seguinte concatenação

$$\begin{aligned} \underbrace{\begin{bmatrix} \mathbf{X}_{(1,p,t,j)} \\ \mathbf{X}_{(2,p,t,j)} \\ \vdots \\ \mathbf{X}_{(I,p,t,j)} \end{bmatrix}}_{\mathbf{X}_{(p,t,j)}} &= \begin{bmatrix} \mathbf{A}'_x D_1(\Phi_x) \\ \mathbf{A}'_x D_2(\Phi_x) \\ \vdots \\ \mathbf{A}'_x D_I(\Phi_x) \end{bmatrix} D_p(\mathbf{W}) D_t(\mathbf{S}^T) D_j(\Phi_z) \mathbf{A}'_z{}^T, \\ \mathbf{X}_{(p,t,j)} &= (\Phi_x \diamond \mathbf{A}'_x) D_p(\mathbf{W}) D_t(\mathbf{S}^T) D_j(\Phi_z) \mathbf{A}'_z{}^T. \end{aligned} \quad (3.85)$$

Em seguida os modos 2 e 3 são agrupados nas colunas realizando a seguinte concatenação para  $j = 1, \dots, J$

$$\begin{aligned} \mathbf{X}_{(p,t)} &= \begin{bmatrix} \mathbf{X}_{(p,t,1)} & \mathbf{X}_{(p,t,2)} & \cdots & \mathbf{X}_{(p,t,J)} \end{bmatrix} \\ &= (\Phi_x \diamond \mathbf{A}'_x) D_p(\mathbf{W}) D_t(\mathbf{S}^T) \begin{bmatrix} D_1(\Phi_z) \mathbf{A}'_z{}^T & D_2(\Phi_z) \mathbf{A}'_z{}^T & \cdots & D_J(\Phi_z) \mathbf{A}'_z{}^T \end{bmatrix} \quad \therefore \end{aligned}$$

$$\mathbf{X}_{(p,t)} = (\Phi_x \diamond \mathbf{A}'_x) D_p(\mathbf{W}) D_t(\mathbf{S}^T) (\Phi_z \diamond \mathbf{A}'_z)^T, \quad (3.86)$$

aplicando a propriedade do produto de Khatri-Rao descrita na equação (3.11) em (3.86) obtemos a seguinte relação

$$\text{vec}(\mathbf{X}_{(p,t)}) = (\Phi_z \diamond \mathbf{A}'_z \diamond \Phi_x \diamond \mathbf{A}'_x) D_p(\mathbf{W}) \text{vecd}(D_t(\mathbf{S}^T)), \quad (3.87)$$

em seguida concatenando os dados recebidos no decorrer dos  $T_s$  *snapshots* que compõem o  $p$ -ésimo sub-bloco obtemos

$$\begin{aligned} \mathbf{X}_{(p)} &= \begin{bmatrix} \text{vec}(\mathbf{X}_{(p,1)}) & \cdots & \text{vec}(\mathbf{X}_{(p,T_s)}) \end{bmatrix} \\ &= (\Phi_z \diamond \mathbf{A}'_z \diamond \Phi_x \diamond \mathbf{A}'_x) D_p(\mathbf{W}) \begin{bmatrix} \text{vecd}(D_1(\mathbf{S}^T)) & \cdots & \text{vecd}(D_{T_s}(\mathbf{S}^T)) \end{bmatrix}, \\ \mathbf{X}_{(p)} &= [\mathcal{H}_2]_{(1)}^T D_p(\mathbf{W}) \mathbf{S}, \quad p = 1, \dots, P. \end{aligned} \quad (3.88)$$

Note que  $[\mathcal{H}_2]_{(1)}^T = (\Phi_z \diamond \mathbf{A}'_z \diamond \Phi_x \diamond \mathbf{A}'_x)$  representa a forma matriciada no modo 1 do seguinte tensor PARAFAC de quinta ordem [9]:

$$\mathcal{H}_2 = \mathcal{I}_{5,M} \times_1 \mathbf{I} \times_2 \mathbf{A}'_x \times_3 \Phi_x \times_4 \mathbf{A}'_z \times_5 \Phi_z \rightarrow [\mathcal{H}_2]_{(1)}^T = (\Phi_z \diamond \mathbf{A}'_z \diamond \Phi_x \diamond \mathbf{A}'_x). \quad (3.89)$$

Finalmente, para obtermos as estruturas desejadas calculamos a partir da equação (3.88) as matrizes de covariância espacial para  $p = 1, \dots, P$  que descrevem os *slices* do tensor  $\bar{\mathcal{R}}$  em termos da decomposição PARATUCK2

$$\bar{\mathbf{R}}_{(p)} \triangleq \frac{1}{T_s} \mathbf{X}_{(p)} \mathbf{X}_{(p)}^H \rightarrow \bar{\mathbf{R}}_{(p)} = \begin{bmatrix} [\mathcal{H}_2]_{(1)}^T \end{bmatrix} D_p(\mathbf{W}) \mathbf{R}_s D_p(\mathbf{W}^*) \begin{bmatrix} [\mathcal{H}_2]_{(1)}^T \end{bmatrix}^H. \quad (3.90)$$

De forma análoga as seções 3.1 e 3.2 obtemos as representações PARAFAC e Tucker4 descritas a seguir:

$$\bar{\mathbf{R}}_{(p)} = \begin{bmatrix} [\mathcal{H}_2]_{(1)}^T \end{bmatrix} D_p(\Omega) \begin{bmatrix} [\mathcal{H}_2]_{(1)}^T \end{bmatrix}^H \rightarrow \text{PARAFAC} \quad (3.91)$$

e

$$\bar{\mathcal{R}} = \mathcal{R}_s \times_1 \begin{bmatrix} [\mathcal{H}_2]_{(1)}^T \end{bmatrix} \times_2 \mathbf{W} \times_3 \begin{bmatrix} [\mathcal{H}_2]_{(1)}^T \end{bmatrix}^* \times_4 \mathbf{W}^* \rightarrow \text{Tucker4}. \quad (3.92)$$

No caso do arranjo planar com *Spatial Smoothing* as matrizes  $\bar{\mathbf{R}}_{\text{mm}}$  e  $\mathbf{A}$  são descritas da seguinte forma:

$$\bar{\mathbf{R}}_{\text{mm}} = (\mathbf{W} \otimes [\mathcal{H}_2]_{(1)}^T) [\mathcal{R}_s]_{(1,2);(3,4)} (\mathbf{W} \otimes [\mathcal{H}_2]_{(1)}^T)^H \quad \text{e} \quad \mathbf{A} = [\mathcal{H}_2]_{(1)}^T. \quad (3.93)$$

A modelagem do receptor para todas as estruturas de arranjos abordadas até o momento recaem nas decomposições PARAFAC, PARATUCK2 e Tucker4. Dessa forma, os algoritmos propostos neste trabalho podem ser utilizados em todos os casos no processo de obtenção da matriz  $\hat{\mathbf{A}}_{(\text{final})}$  e o estudo de

identificabilidade fornecido na Seção 3.4 pode ser facilmente estendido para os casos com *Spatial Smoothing*. Considerando todas as configurações de arranjos utilizadas até o momento, estimções para as matrizes de assinaturas espaciais  $\hat{A}_x$ ,  $\hat{A}_y$  e  $\hat{A}_z$  e para as matrizes de defasagem  $\hat{\Phi}_x$ ,  $\hat{\Phi}_y$  e  $\hat{\Phi}_z$  podem ser obtidas a partir de  $\hat{A}_{(\text{final})}$  de acordo com a Tabela 3.1:

**Tabela 3.1:** Estrutura da Matriz  $\hat{A}_{(\text{final})}$ .

Configuração do Arranjo	$\hat{A}_{(\text{final})}$
<i>2L-Shape</i>	$\begin{bmatrix} \hat{A}_x \\ \hat{A}_y \\ \hat{A}_z \end{bmatrix}$
<i>2L-Shape com Spatial Smoothing</i>	$\begin{bmatrix} (\hat{\Phi}_x \diamond \hat{A}'_x) \\ (\hat{\Phi}_y \diamond \hat{A}'_y) \\ (\hat{\Phi}_z \diamond \hat{A}'_z) \end{bmatrix}$
Planar	$(\hat{A}_z \diamond \hat{A}_x)$
Planar com <i>Spatial Smoothing</i>	$(\hat{\Phi}_z \diamond \hat{A}'_z \diamond \hat{\Phi}_x \diamond \hat{A}'_x)$

Note que apenas na configuração *2L-Shape* sem *Spatial Smoothing* é possível obter estimções diretas para as matrizes  $\hat{A}_x$ ,  $\hat{A}_y$  e  $\hat{A}_z$  realizando o particionamento de  $\hat{A}_{(\text{final})}$ . Entretanto, nas demais configurações não é possível obtê-las diretamente, pois sua estrutura está relacionada com o produto de Khatri-Rao entre as matrizes de interesse. Nesses casos, a utilização do algoritmo LS-KRF como último passo dos estimadores propostos torna-se necessária e sua descrição será realizada detalhadamente a seguir:

### 3.9 Least-Squares Khatri-Rao Factorization (LS-KRF)

O LS-KRF (do inglês, *Least-Squares Khatri-Rao Factorization*) foi proposto em [67] e vêm sendo amplamente utilizado na área de processamento de sinais, principalmente em trabalhos que envolvem estimação de parâmetros em sistemas de comunicação [8] e [5]. Sua utilização proporciona solucionar dois tipos de problemas, o primeiro quando uma matriz é conhecida e o segundo quando ambas as matrizes são desconhecidas [68]. No decorrer deste trabalho a segunda abordagem será utilizada, detalhes sobre a primeira podem ser obtidos em [68]. De forma arbitrária, vamos considerar que a matriz  $C \in \mathbb{C}^{MN \times P}$  é descrita pelo produto de Khatri-Rao entre as matrizes  $A \in \mathbb{C}^{M \times P}$  e  $B \in \mathbb{C}^{N \times P}$ , ou seja  $C = A \diamond B$ , e desejamos obter as estimções de  $\hat{A}$  e  $\hat{B}$  que minimizam a seguinte função custo:

$$f_{\text{custo}}(\hat{A}, \hat{B}) = \left\| C - \hat{A} \diamond \hat{B} \right\|_{\text{F}}^2. \quad (3.94)$$

A partir da definição do produto de Khatri-Rao descrita em (2.41) a  $p$ -ésima coluna de  $C$ , descrita por  $c_p$ , é obtida através do produto de Kronecker entre a  $p$ -ésima coluna das matrizes  $A$  e  $B$ , ou seja  $c_p = a_p \otimes b_p$ , de forma que se os dados contidos em  $c_p$  forem reorganizados como uma matriz  $\tilde{C}_p \in \mathbb{C}^{N \times M}$  em que  $\text{vec}(\tilde{C}_p) = c_p$  então  $\tilde{C}_p$  pode ser reescrita da seguinte forma

$$\tilde{C}_p = b_p \cdot a_p^T. \quad (3.95)$$

Estimações para a  $p$ -ésima coluna das matrizes  $\hat{A}$  e  $\hat{B}$  podem ser obtidas calculando a SVD (do inglês, *Singular Value Decomposition*) da matriz  $\tilde{C}_p$ , ou seja,

$$\tilde{C}_p = U_p \Sigma_p V_p^H, \quad (3.96)$$

de forma que a melhor aproximação de  $\tilde{C}_p$  como uma matriz de *rank-1* é obtida através do truncamento da SVD em seu primeiro termo e as estimações para a  $p$ -ésima coluna das matrizes  $\hat{A}$  e  $\hat{B}$  são descritas por:

$$\hat{a}_p = \sqrt{\sigma_1} \cdot v_1^* \in \mathbb{C}^{M \times 1}, \quad (3.97)$$

e

$$\hat{b}_p = \sqrt{\sigma_1} \cdot u_1 \in \mathbb{C}^{N \times 1}, \quad (3.98)$$

em que  $u_1$  e  $v_1$  são vetores formados pela primeira coluna das matrizes  $U_p$  e  $V_p$ , respectivamente, e  $\sigma_1$  é o maior valor singular extraído da matriz  $\Sigma_p$  para  $p = 1, \dots, P$ . A seguir o algoritmo LS-KRF será apresentado de forma resumida:

### Algoritmo LS-KRF

---

Fixe  $p = 1$ ;

1. Reorganize os dados contidos em  $c_p$  como uma matriz  $\tilde{C}_p \in \mathbb{C}^{N \times M}$  de forma que  $\text{vec}(\tilde{C}_p) = c_p$ ;
2. Obtenha as matrizes  $U_p$ ,  $\Sigma_p$  e  $V_p$  a partir da SVD de  $\tilde{C}_p$ :

$$\tilde{C}_p = U_p \Sigma_p V_p^H$$

3. Calcule as estimações para a  $p$ -ésima coluna de  $\hat{A}$  e  $\hat{B}$ :

$$\hat{A}(:, p) = \sqrt{\sigma_1} \cdot v_1^* \quad \text{e} \quad \hat{B}(:, p) = \sqrt{\sigma_1} \cdot u_1$$

em que  $v_1 = V_p(:, 1)$  e  $u_1 = U_p(:, 1)$  e  $\sigma_1$  é o maior valor singular de  $\Sigma_p$ ;

4. Se  $p < P$ , fixe  $p = p + 1$  e repita os passos 1-3.
-

Para os arranjos  $2L$ -Shape com *Spatial Smoothing* e planar a forma padrão do algoritmo apresentada anteriormente é aplicada diretamente nas matrizes  $(\hat{\Phi}_x \diamond \hat{A}'_x)$ ,  $(\hat{\Phi}_y \diamond \hat{A}'_y)$ ,  $(\hat{\Phi}_z \diamond \hat{A}'_z)$  e  $(\hat{A}_z \diamond \hat{A}_x)$ , respectivamente. Entretanto, no caso do arranjo planar com *Spatial Smoothing* a generalização do LS-KRF para o caso  $\mathbf{A}^{(1)} \diamond \mathbf{A}^{(2)} \diamond \dots \diamond \mathbf{A}^{(N)}$  sendo aplicado  $N - 1$  vezes consecutivas possibilita realizar as estimações a partir de  $(\hat{\Phi}_z \diamond \hat{A}'_z \diamond \hat{\Phi}_x \diamond \hat{A}'_x)$ .

É importante salientar que as soluções obtidas utilizando o LS-KRF não são únicas pois as colunas das matrizes estimadas são multiplicadas por um fator de escala. Entretanto, como em todos os casos a primeira linha das matrizes originais são conhecidas com elementos iguais a 1 esta ambiguidade pode ser facilmente removida e os ângulos  $\hat{\alpha}_m$  e  $\hat{\beta}_m$  podem ser obtidos independentemente a partir de  $\hat{A}'_x$ ,  $\hat{A}'_y$  e  $\hat{A}'_z$  de acordo com as equações (3.54), (3.55) e (3.56) e a partir de  $\hat{\Phi}_x$ ,  $\hat{\Phi}_y$  e  $\hat{\Phi}_z$  utilizando as equações (2.25), (2.30) e (2.31) de modo que as estimações finais representam a média aritmética dos valores estimados em cada caso.

## Resultados Numéricos e Discussões

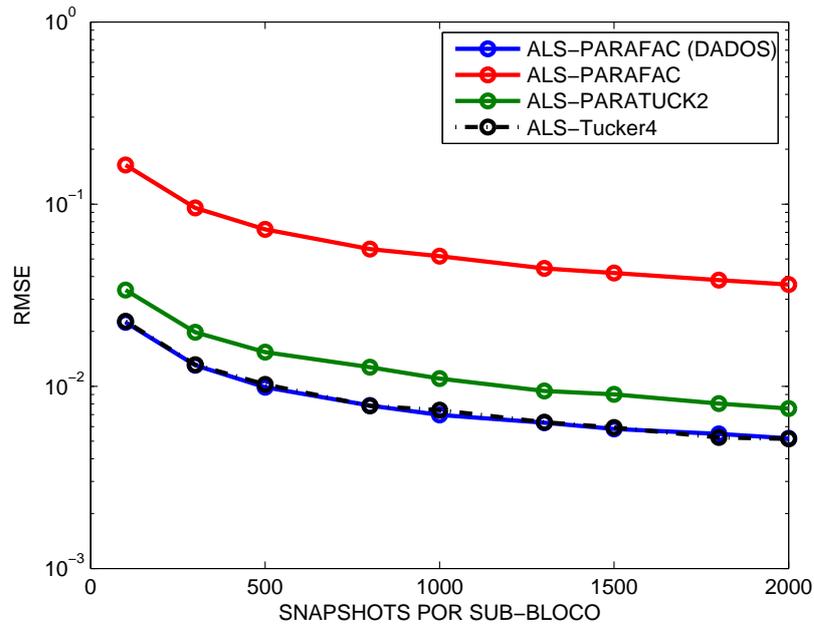
Neste capítulo, os resultados de simulação obtidos em alguns exemplos numéricos serão apresentados com o objetivo de avaliar o desempenho e validar os métodos propostos neste trabalho. Esta análise é realizada em função da **SNR** do sistema para cada um dos métodos apresentados anteriormente considerando vários cenários constituídos a partir da variação do número de fontes, *snapshots* e sensores que compõem o arranjo. Os resultados obtidos serão apresentados e discutidos separadamente para cada configuração de arranjo utilizada, entretanto todas as simulações refletem o comportamento médio obtido a partir de 1000 simulações de Monte Carlo independentes. Em cada simulação os símbolos transmitidos seguem uma modulação **BPSK** (do inglês, *Binary Phase-Shift Keying*), a matriz de potências  $W$  é assumida conhecida no receptor e segue uma distribuição uniforme e a distância entre os sensores é considerada fixa e igual a  $\lambda/2$ .

### 4.1 Resultados para o Arranjo Linear Uniforme

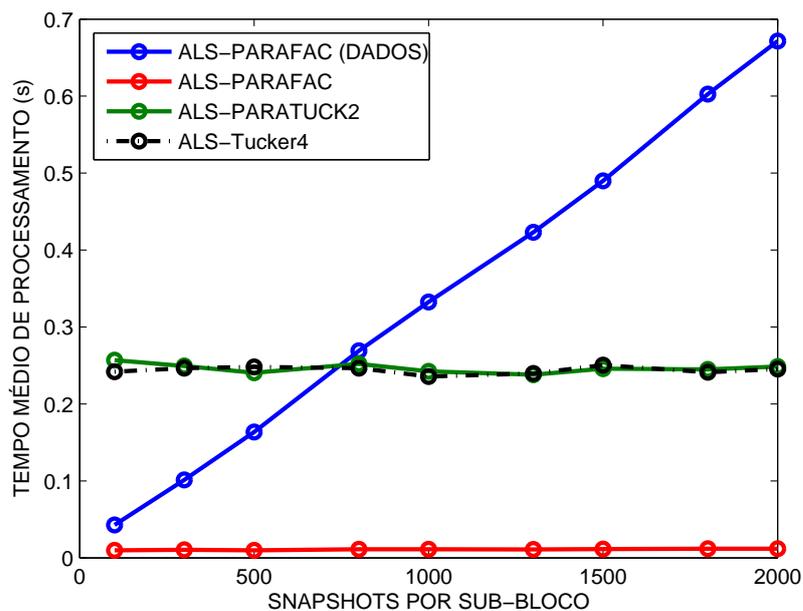
Nesta seção serão apresentados os resultados obtidos para o arranjo linear uniforme. A primeira análise realizada descreve o desempenho da estimação dos ângulos de elevação das fontes obtidos a partir de  $\hat{A}_{(\text{final})}$ . A métrica utilizada neste primeiro caso é o **RMSE** (do inglês, *Normalized Mean Square Error*) definido para uma dada **SNR** da seguinte forma:

$$\text{RMSE}_{(\text{SNR})} = \frac{1}{1000} \sum_{w=1}^{1000} \sqrt{\frac{1}{M} \sum_{m=1}^M \left( \alpha_m - \hat{\alpha}_m^{(w)} \right)^2}, \quad (4.1)$$

em que  $\hat{\alpha}_m^{(w)}$  representa a estimação do ângulo da  $m$ -ésima fonte na  $w$ -ésima simulação de Monte Carlo. Na Figura 4.1 é considerado um cenário com

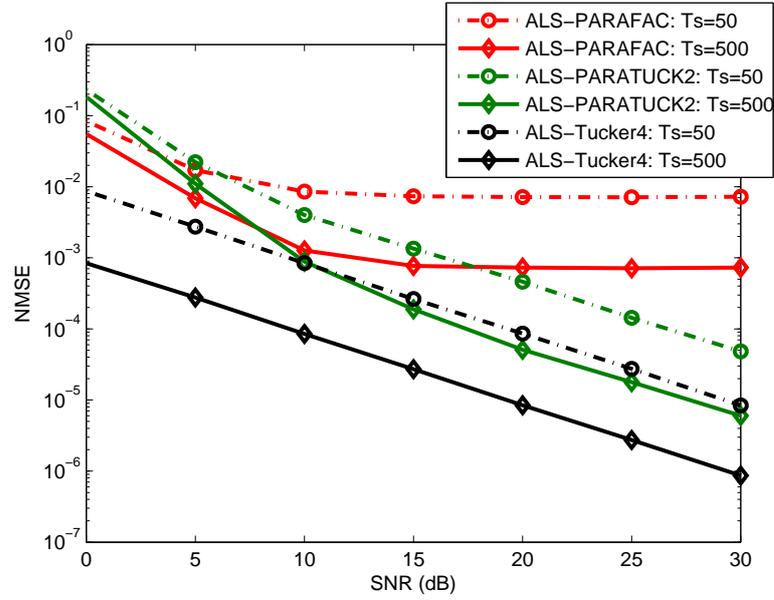


**Figura 4.1:** RMSE *versus*  $T_s$  para  $M = 2, N = 5$  e  $P = 20$  para SNR = 15 dB e configuração ULA no receptor.



**Figura 4.2:** Tempo médio de processamento *versus*  $T_s$  para  $M = 2, N = 5$  e  $P = 20$  para SNR = 15 dB e configuração ULA no receptor.

$M = 2, N = 5$  e  $P = 20$  para uma SNR fixa e igual à 15 dB. A partir deste experimento, é possível perceber que o método proposto ALS-Tucker4 apresenta um desempenho em termos do RMSE semelhante ao ALS-PARAFAC aplicado diretamente no tensor de dados. Além disso, ambos apresentam desempenhos superiores aos métodos ALS-PARAFAC e ALS-PARATUCK2 baseados em covariâncias espaciais. No que diz respeito ao tempo médio



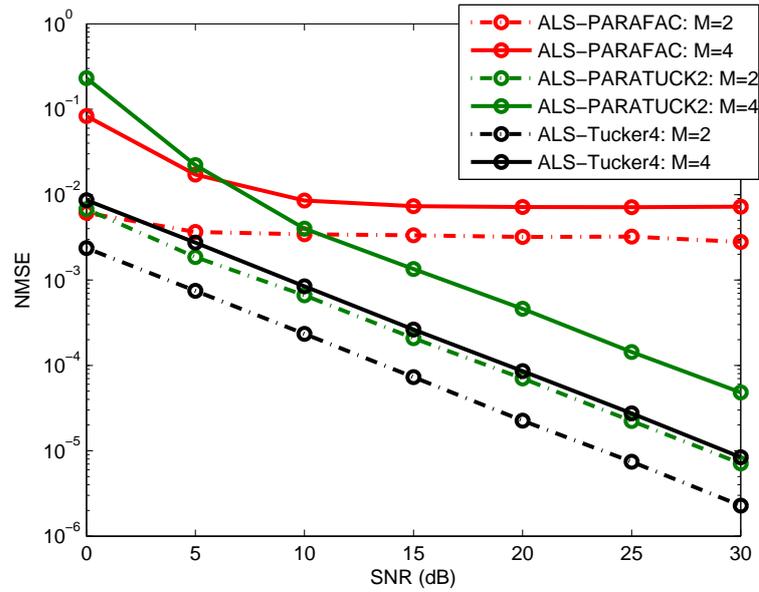
**Figura 4.3:** NMSE de  $\hat{A}_{(\text{final})}$  versus SNR para  $M = 4$ ,  $N = 3$  e  $P = 25$  considerando diferentes valores de  $T_s$  e configuração ULA no receptor.

de processamento nota-se, a partir da Figura 4.2, que os métodos propostos apresentam menor complexidade computacional em relação a aplicação do ALS-PARAFAC diretamente no tensor de dados quando valores elevados de *snapshots* por sub-bloco são considerados. Este comportamento é proveniente da independência em relação à  $T_s$  obtida após o cálculo das covariâncias espaciais e espaço-temporais dos dados recebidos.

As próximas análises descrevem o desempenho da estimação da matriz de assinaturas espaciais utilizando como métrica o **NMSE** (do inglês, *Normalized Mean Square Error*) em função da **SNR**, sendo este definido como:

$$\text{NMSE}_{(\text{SNR})} = \frac{1}{1000} \sum_{w=1}^{1000} \frac{\|\hat{A}_{(\text{final})}^{(w)} - \mathbf{A}\|_F^2}{\|\mathbf{A}\|_F^2}, \quad (4.2)$$

em que  $\hat{A}_{(\text{final})}^{(w)}$  representa a estimação de  $\mathbf{A}$  obtida na  $w$ -ésima simulação de Monte Carlo. A Figura 4.3 apresenta o desempenho em termos de **NMSE** para diferentes valores de  $T_s$  considerando um cenário com  $M = 4$ ,  $N = 3$  e  $P = 25$  sendo os ângulos de elevação das fontes iguais a  $\alpha_1 = 21.2^\circ$ ,  $\alpha_2 = 37.5^\circ$ ,  $\alpha_3 = 56.3^\circ$  e  $\alpha_4 = 77.4^\circ$ . Pode ser visto que os métodos propostos ALS-PARATUCK2 e ALS-Tucker4 superam o ALS-PARAFAC em todas as situações. O melhor desempenho obtido pelo ALS-Tucker4 está relacionado à estrutura tensorial de quarta ordem obtida em sua modelagem que proporciona uma melhor utilização da estrutura multilinear dos dados. O piso exibido pelo método ALS-PARAFAC está diretamente relacionado com o erro de modelagem devido à suposição de que a matriz de covariância das fontes seja perfeitamente

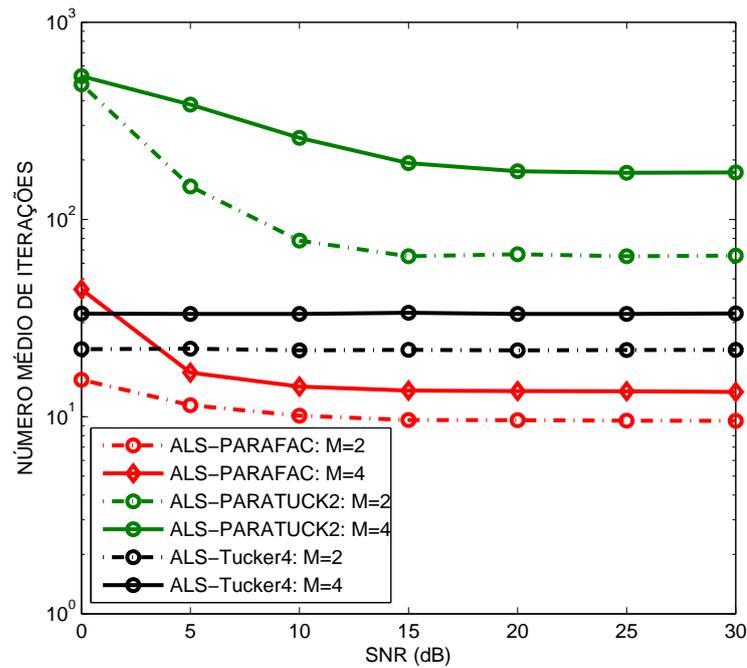


**Figura 4.4:** NMSE de  $\hat{A}_{(\text{final})}$  versus SNR para  $N = 3$ ,  $T_s = 50$  e  $P = 25$  considerando diferentes valores de  $M$  e configuração ULA no receptor.

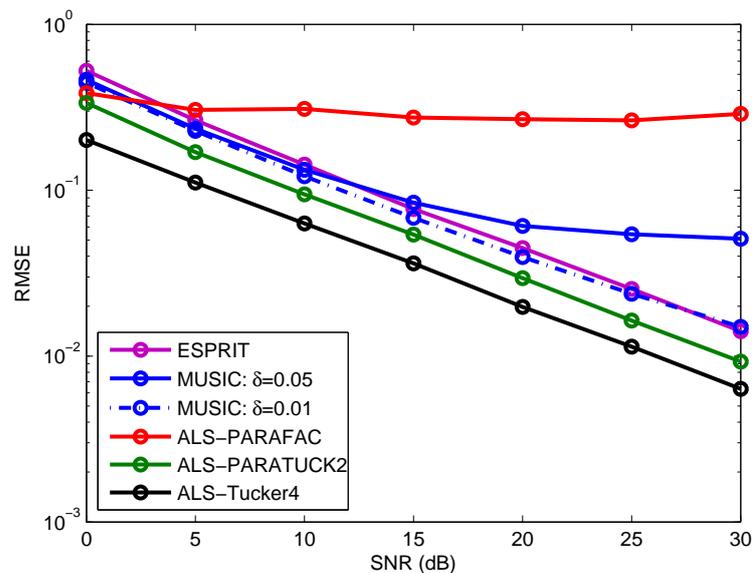
diagonal. Portanto, quando se trabalha com covariâncias de amostras reais os métodos propostos são preferíveis.

Na Figura 4.4 o desempenho é avaliado para diferentes valores de  $M$  considerando um cenário em que  $N = 3$ ,  $T_s = 50$  e  $P = 25$ . Nos casos em que  $M = 4$  os mesmos ângulos de elevação considerados na simulação anterior foram utilizados, entretanto para  $M = 2$  foram considerados  $\alpha_1 = 37.5^\circ$  e  $\alpha_2 = 77.4^\circ$ . Note que, com o aumento do número de fontes e fixando o número de sensores todos os métodos apresentaram uma degradação na qualidade da estimação, visível com o aumento do NMSE para uma dada SNR. Nesse caso, percebe-se que o método ALS-PARATUCK2 apresenta maior sensibilidade em comparação ao ALS-Tucker4. Entretanto, esta degradação não promove uma perda acentuada na qualidade da estimação, de modo que estes métodos podem ser aplicados confiavelmente nesse tipo de cenário. Assim como na simulação anterior os métodos propostos superaram o ALS-PARAFAC em virtude do erro de modelagem existente neste método.

Como os métodos propostos são iterativos uma métrica indispensável na análise do desempenho é o número de iterações necessárias para a convergência dos algoritmos. Na Figura 4.5 o desempenho é avaliado em termos do número médio de iterações em função da SNR utilizando os mesmos parâmetros da Figura 4.4. Em todos os casos o número de iterações aumenta quando um número maior de fontes é considerado. Note que o ALS-PARATUCK2 converge com mais iterações em relação aos demais métodos enquanto o número de iterações no ALS-Tucker4 permanece constante



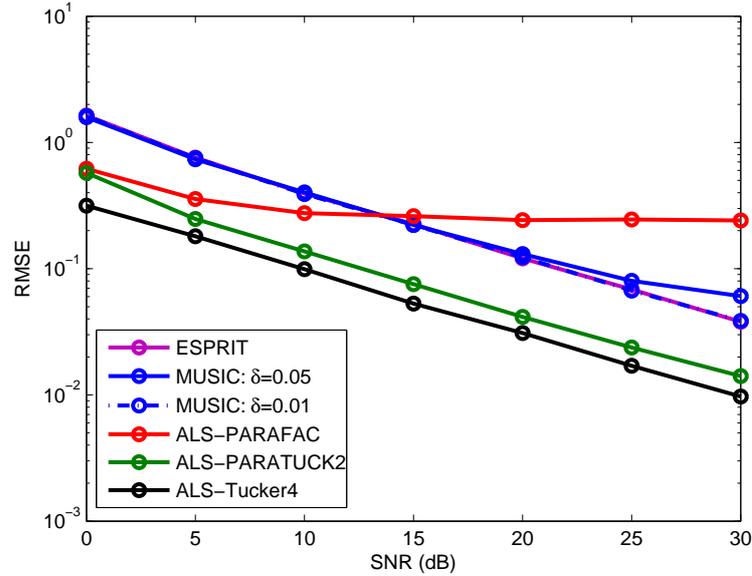
**Figura 4.5:** Número médio de iterações *versus* SNR para  $N = 3$ ,  $T_s = 50$  e  $P = 25$  considerando diferentes valores de  $M$  e configuração ULA no receptor.



**Figura 4.6:** RMSE *versus* SNR para  $M = 2$ ,  $N = 6$ ,  $T_s = 40$  e  $P = 20$  e configuração ULA no receptor.

independente da SNR. Pode ser visto também que o método ALS-PARAFAC possui o menor número de iterações em todos os experimentos, entretanto sua utilização é inviável como mostra a Figura 4.4.

A Figura 4.6 mostra o RMSE das estimações dos ângulos de elevação em função da SNR para um cenário com  $M = 2$ ,  $N = 6$ ,  $T_s = 40$  e  $P = 20$  e ângulos  $\alpha_1 = 28.2^\circ$  e  $\alpha_2 = 54.3^\circ$ . Neste experimento os desempenhos são comparados

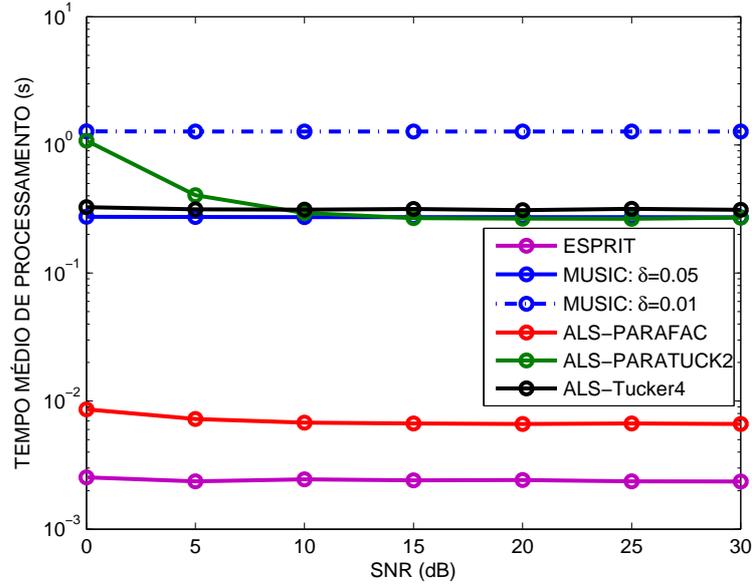


**Figura 4.7:** RMSE versus SNR para  $M = 2$ ,  $N = 4$ ,  $T_s = 40$  e  $P = 20$  e configuração ULA no receptor.

aos métodos matriciais MUSIC com  $\delta = 0.01$  e  $\delta = 0.05$  e ESPRIT. O parâmetro  $\delta$  representa o intervalo (passo) entre os ângulos de elevação utilizados na varredura do espectro  $P_{\text{MUSIC}}$ . Note que os métodos propostos apresentam melhor desempenho em relação ao MUSIC e ESPRIT sob as mesmas condições de simulação sendo mais acentuado no ALS-Tucker4 em todos os valores de SNR considerados. O piso exibido pelo método MUSIC com  $\delta = 0.05$  está relacionado com a utilização de um passo elevado na varredura.

Na Figura 4.7 um cenário mais desafiador é considerado, ou seja, o número de sensores é reduzido para  $N = 4$  em comparação a simulação anterior. Nesse experimento os métodos propostos ainda apresentam melhor desempenho em comparação aos métodos concorrentes ALS-PARAFAC, MUSIC e ESPRIT. Entretanto, os ganhos obtidos pelos métodos tensoriais propostos são mais evidenciados em comparação aos resultados observados na Figura 4.6 tornando-os mais atraentes principalmente quando um número reduzido de sensores é utilizado no arranjo receptor.

A Figura 4.8 mostra o tempo médio de processamento em segundos em função da SNR considerando o mesmo cenário descrito na Figura 4.6. Note que os métodos ALS-PARATUCK2 e ALS-Tucker4 apresentam um tempo de processamento semelhante ao método MUSIC com  $\delta = 0.05$  para valores médios e altos de SNR. Em comparação ao MUSIC utilizando  $\delta = 0.01$  os métodos tensoriais apresentam desempenho superior em virtude do pequeno valor de  $\delta$  utilizado que acarreta numa busca exaustiva para um grande número de pontos que compõem o espectro espacial  $P_{\text{MUSIC}}$ . Com relação



**Figura 4.8:** Tempo médio de processamento *versus* SNR para  $M = 2$ ,  $N = 6$ ,  $T_s = 40$  e  $P = 20$  e configuração ULA no receptor.

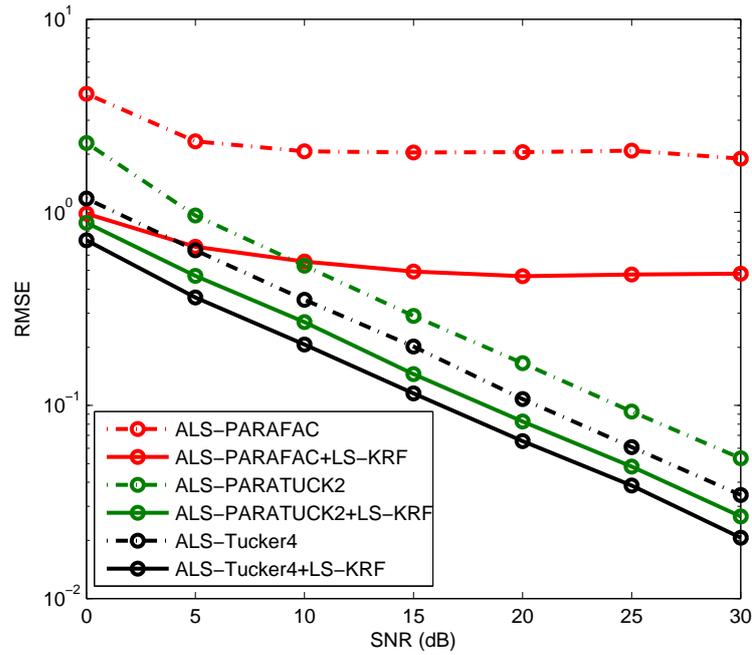
ao ESPRIT o desempenho torna-se substancialmente inferior em virtude desse método apresentar uma solução fechada diferentemente dos métodos propostos que são baseados em algoritmos iterativos.

## 4.2 Resultados para o Arranjo 2L-Shape

A análise para os arranjos 2L-Shape e Planar será realizada considerando o par elevação e azimute simultaneamente no cálculo do RMSE. Esta métrica é denominada RMSE total dos ângulos estimados definida da seguinte forma para uma dada SNR:

$$\text{RMSE}_{(\text{SNR})} = \frac{1}{1000} \sum_{w=1}^{1000} \sqrt{\frac{1}{M} \sum_{m=1}^M \left[ \left( \alpha_m - \hat{\alpha}_m^{(w)} \right)^2 + \left( \beta_m - \hat{\beta}_m^{(w)} \right)^2 \right]}, \quad (4.3)$$

em que  $\hat{\beta}_m^{(w)}$  é a estimativa do ângulo de elevação da  $m$ -ésima fonte na  $w$ -ésima simulação de Monte Carlo. A Figura 4.9 representa um cenário em que  $M = 3$ ,  $P = 25$ ,  $T_s = 40$  e o número de sensores em cada eixo  $N_x$ ,  $N_y$  e  $N_z$  são iguais a 4. Os pares  $(\alpha_m, \beta_m)$  são dados por  $(22.4^\circ, 31.6^\circ)$ ,  $(39.2^\circ, 50.5^\circ)$  e  $(72.5^\circ, 65.8^\circ)$  para  $m = 1, \dots, 3$ , respectivamente. Observe que neste experimento os métodos ALS-PARATUCK2 e ALS-Tucker4 adicionados do LS-KRF apresentaram um desempenho superior em comparação aos métodos de recepção padrão do arranjo 2L-Shape para todos os valores de SNR considerados. A utilização da técnica *Spatial Smoothing* dividindo cada arranjo em dois sub-arranjos idênticos torna-se uma opção atraente que proporciona um ganho acentuado

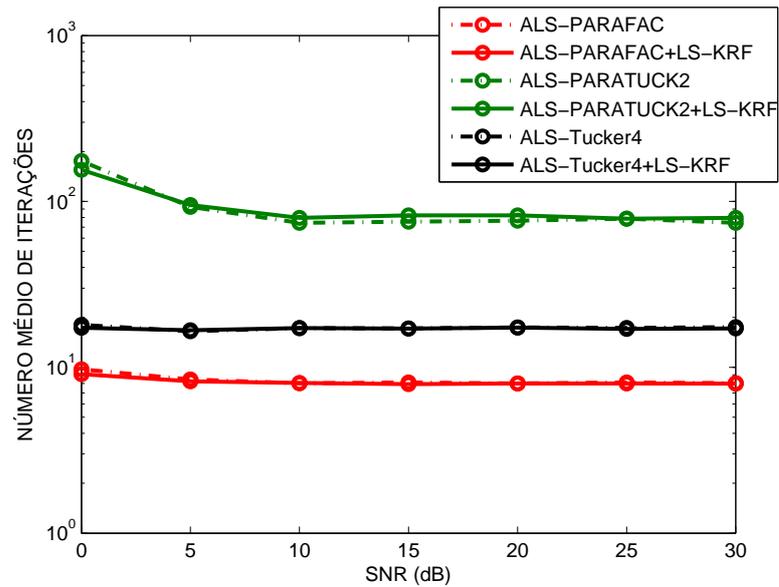


**Figura 4.9:** RMSE versus SNR para  $M = 3$ ,  $P = 25$ ,  $T_s = 40$  e  $N_x, N_y$  e  $N_z$  iguais a 4 e configuração 2L-Shape no receptor.

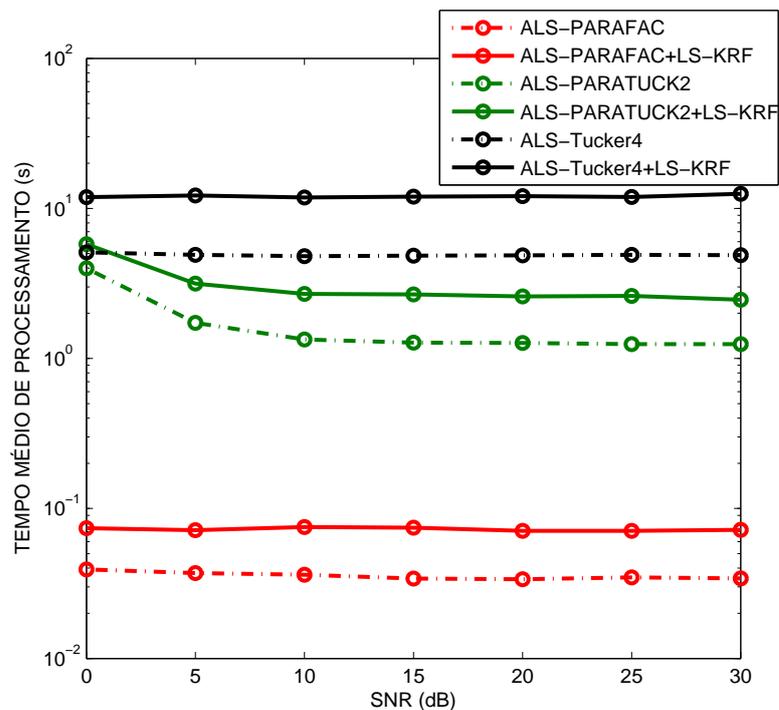
na qualidade da estimação da DoA mesmo para cenários que possuem um número reduzido de sensores em cada eixo. Além disso, esta técnica descarta a necessidade de utilizar uma quantidade maior de sensores pois um comportamento equivalente é obtido a partir do aumento da diversidade espacial oriunda da propriedade de invariância ao deslocamento existente nesse tipo de arranjo.

Nas Figuras 4.10 e 4.11 o mesmo cenário descrito no experimento anterior é considerado. Nestas simulações a análise do desempenho é realizada em termos do número médio de iterações e do tempo médio de processamento. Na Figura 4.10 note que o número de iterações independe da técnica utilizada e assim como observado na configuração ULA o método ALS-PARATUCK2 converge com mais iterações em relação aos demais métodos tensoriais. A Figura 4.11 mostra o tempo médio de processamento de cada um dos métodos. Observe que o ganho alcançado em termos de RMSE com a utilização da técnica *Spatial Smoothing* reflete num aumento do tempo médio de processamento. Este comportamento é proveniente do aumento do número de sensores utilizados na recepção dos sinais que geram naturalmente um aumento nas dimensões do tensor de dados, afetando diretamente no tempo de processamento das operações matemáticas intrínsecas nos algoritmos.

Na Figura 4.12 o desempenho dos métodos de estimação tensoriais utilizando *Spatial Smoothing* são comparados aos métodos matriciais MUSIC

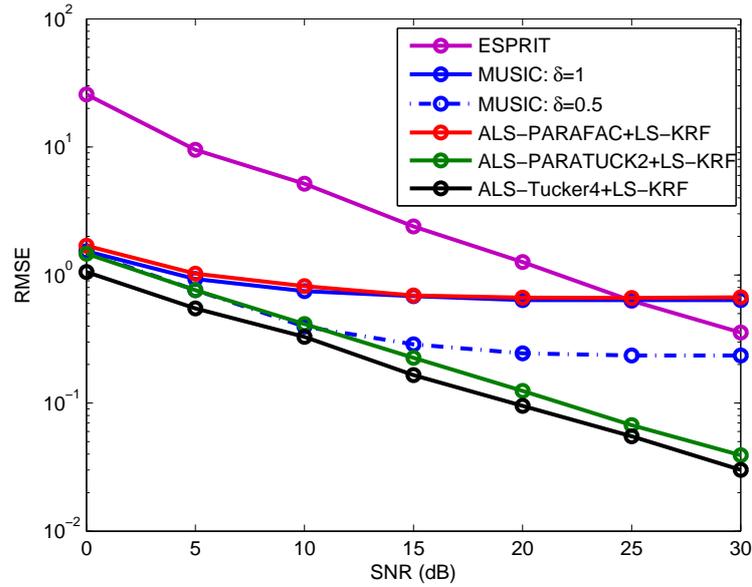


**Figura 4.10:** Número médio de iterações *versus* SNR para  $M = 3, P = 25, T_s = 40$  e  $N_x, N_y$  e  $N_z$  iguais a 4 e configuração *2L-Shape* no receptor.



**Figura 4.11:** Tempo médio de processamento *versus* SNR para  $M = 3, P = 25, T_s = 40$  e  $N_x, N_y$  e  $N_z$  iguais a 4 e configuração *2L-Shape* no receptor.

e **ESPRIT** para o arranjo *2L-Shape* em termos do RMSE total considerando um cenário com os seguintes parâmetros:  $M = 2, P = 25, T_s = 40$  e  $N_x, N_y$  e  $N_z$  iguais a 3 e as direções de chegada das fontes são dadas pelos seguintes pares de azimute e elevação  $(22.4^\circ, 31.6^\circ)$  e  $(39.2^\circ, 50.5^\circ)$ . Nesse experimento os métodos ALS-PARATUCK2 e ALS-Tucker4 apresentam um desempenho muito superior em comparação ao **ESPRIT**. Esse ganho acentuado está

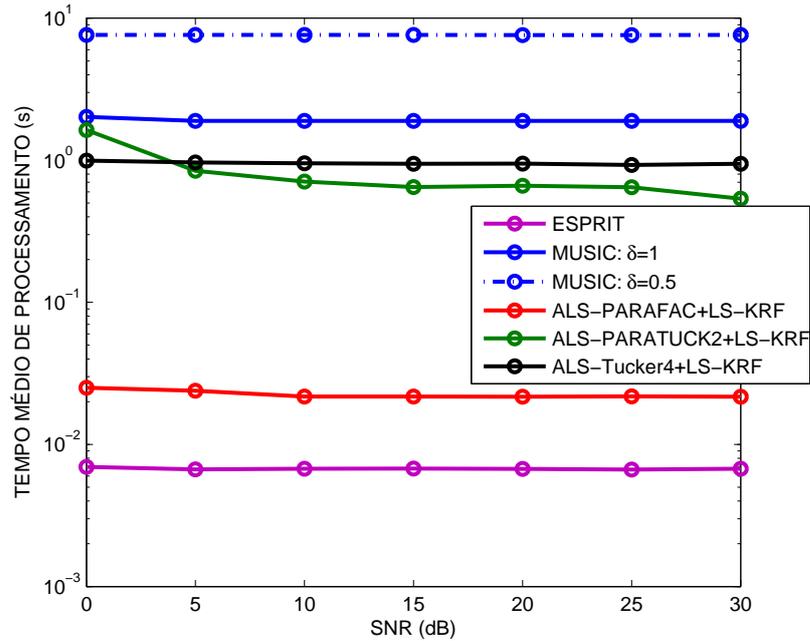


**Figura 4.12:** RMSE versus SNR para  $M = 2, P = 25, T_s = 40$  e  $N_x, N_y$  e  $N_z$  iguais a 3 e configuração *2L-Shape* no receptor.

relacionado as limitações apresentadas por esse método provenientes do fato das estimações dos ângulos de elevação e azimute serem realizadas utilizando cada eixo separadamente conforme descrito no Capítulo 2, enquanto que nos demais métodos a geometria *2L-Shape* é utilizada completamente.

Em comparação ao **MUSIC** os métodos propostos apresentam um ganho considerável para valores de SNR superiores à 10dB quando  $\delta = 1$ , caso em que o passo utilizado na varredura do espectro espacial é extremamente elevado. Diminuindo o passo para  $\delta = 0.5$  observa-se que o desempenho do método ALS-PARTUCK2 é praticamente igual ao **MUSIC** para baixos valores de SNR podendo ser considerado igual para toda a faixa de SNR simulada quando um passo extremamente pequeno for utilizado. Entretanto, para valores elevados de SNR os métodos tensoriais propostos apresentam melhor desempenho pois o RMSE decresce linearmente enquanto que no **MUSIC** ocorre uma saturação a partir de 15dB proveniente do valor de  $\delta$  utilizado na simulação.

Na Figura 4.13 a análise do desempenho da simulação anterior é realizada utilizando o tempo médio de processamento como métrica. Observe que os métodos ALS-PARATUCK2 e ALS-Tucker4 possuem um tempo de processamento inferior ao **MUSIC** tornando-os mais viáveis principalmente quando  $\delta = 0.5$ , situação em que os métodos propostos apresentam um tempo médio de processamento aproximadamente dez vezes menor que o **MUSIC** mesmo para valores de SNR em que o RMSE são próximos como mostra a Figura 4.12. Para  $\delta = 1$  a utilização do **MUSIC** pode ser descartada pois seu desempenho é insatisfatório em termos de RMSE e tempo médio de



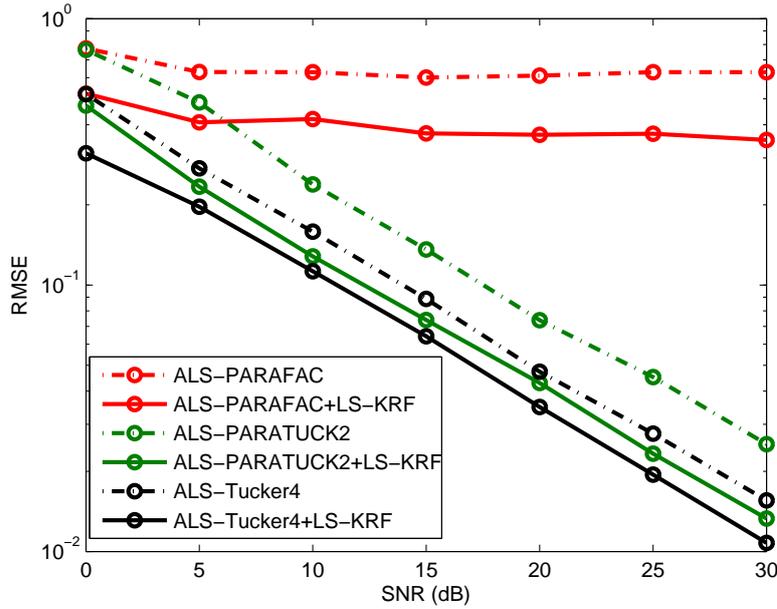
**Figura 4.13:** Tempo médio de processamento *versus* SNR para  $M = 2, P = 25, T_s = 40$  e  $N_x, N_y$  e  $N_z$  iguais a 3 e configuração *2L-Shape* no receptor.

processamento. Como visto na Figura 4.12 quando o valor de  $\delta$  diminui o desempenho do MUSIC melhora consideravelmente, em contrapartida, nesse caso o tempo médio de processamento aumenta consideravelmente em virtude do grande número de pontos utilizados na composição do espectro  $P_{\text{MUSIC}}$ .

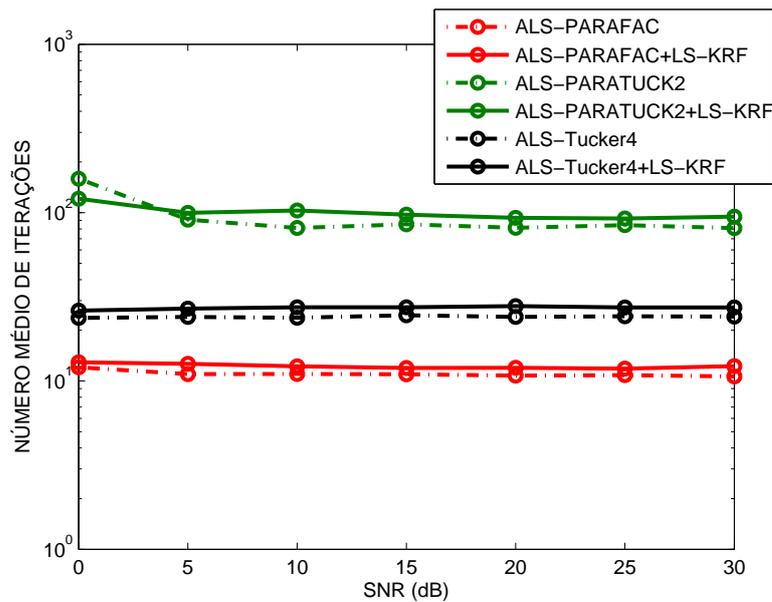
### 4.3 Resultados para o Arranjo Planar

Os resultados obtidos referentes ao arranjo de sensores planar descrevem o comportamento médio de um cenário em que  $M = 2, P = 25, T_s = 50$  e  $N_x$  e  $N_z$  são iguais a 4, totalizando 16 sensores uniformemente distribuídos no plano  $x - z$ . Os pares  $(\alpha_m, \beta_m)$  são dados por  $(25.4^\circ, 31.3^\circ)$  e  $(76.2^\circ, 70.5^\circ)$ . Note que na Figura 4.14 de forma semelhante a configuração *2L-Shape*, a utilização da técnica *Spatial Smoothing* dividindo o arranjo planar em dois sub-arranjos idênticos compostos por 9 sensores cada, também apresenta um desempenho superior em termos de RMSE em comparação aos métodos ALS-PARATUCK2 e ALS-Tucker4 padrão.

Nesse caso, é possível perceber que uma maior variação é obtida quando o receptor é modelado via decomposição PARATUCK2, entretanto um melhor desempenho é alcançado quando o ALS-Tucker4 adicionado do LS-KRF é utilizado. Da mesma forma que nos demais casos, a utilização do ALS-PARAFAC torna-se inviável em virtude da não diagonalidade da matriz  $R_s$ . Em relação ao número médio de iterações para a convergência, a Figura 4.15 mostra que mesmo com o aumento do número de dados recebidos no



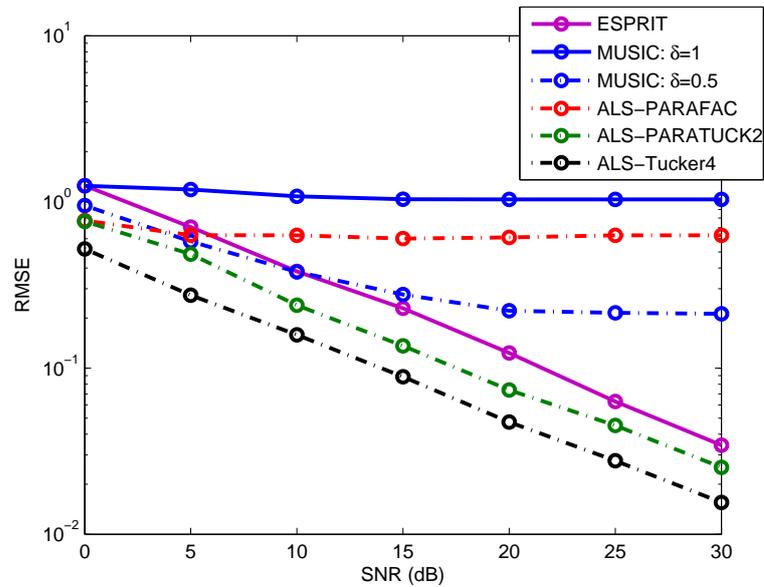
**Figura 4.14:** RMSE *versus* SNR para  $M = 2, P = 25, T_s = 50$  e  $N_x$  e  $N_z$  iguais a 4 e configuração Planar no receptor.



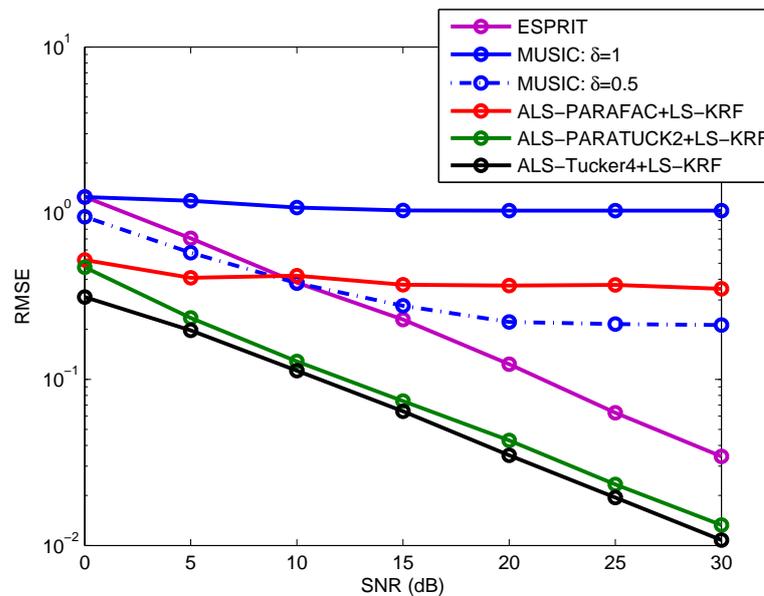
**Figura 4.15:** Número médio de iterações *versus* SNR para  $M = 2, P = 25, T_s = 50$  e  $N_x$  e  $N_z$  iguais a 4 e configuração planar no receptor.

arranjo de sensores proveniente da utilização da técnica *Spatial Smoothing* uma variação considerável no número de iterações não é observada.

Nas Figuras 4.16 e 4.17 o desempenho dos métodos propostos são comparados com o MUSIC e ESPRIT. Na Figura 4.16 a propriedade de invariância ao deslocamento não foi utilizada de modo que as estimações foram obtidas diretamente a partir do ALS-PARAFAC, ALS-PARATUCK2 e ALS-Tucker4. Note que neste caso os métodos propostos apresentam desempenho superior em relação ao MUSIC e ESPRIT. Entretanto, o ganho

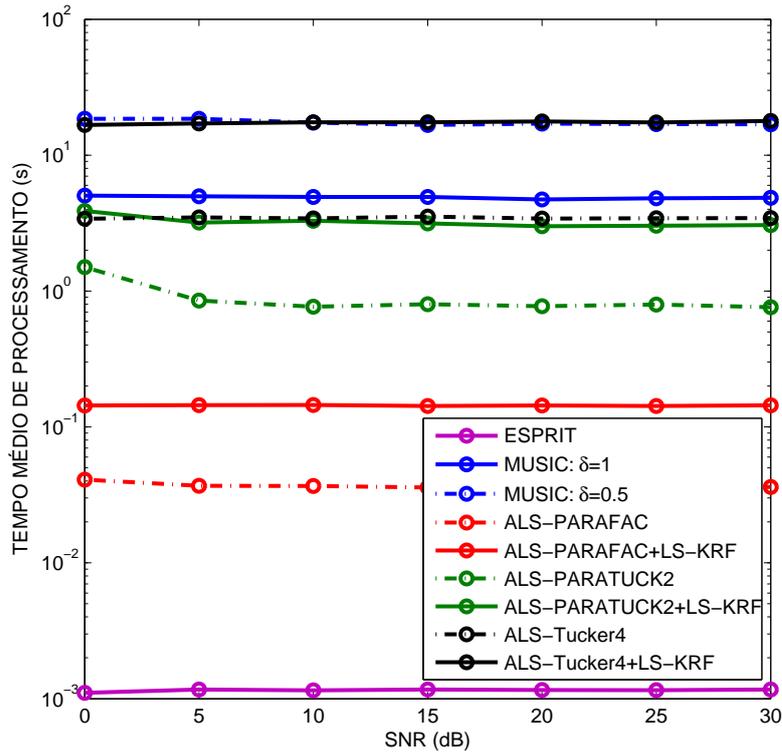


**Figura 4.16:** RMSE versus SNR para  $M = 2, P = 25, T_s = 50$  e  $N_x$  e  $N_z$  iguais a 4 e configuração Planar sem *Spatial Smoothing*.



**Figura 4.17:** RMSE versus SNR para  $M = 2, P = 25, T_s = 50$  e  $N_x$  e  $N_z$  iguais a 4 e configuração Planar com *Spatial Smoothing*.

obtido é praticamente constante em relação ao ESPRIT para toda a faixa de SNR utilizada. Em relação ao MUSIC o ganho torna-se mais acentuado para valores elevados de SNR em virtude do piso exibido por este método provocado pelos valores de  $\delta$  utilizados. Na Figura 4.17 pode ser visto que a utilização da técnica *Spatial Smoothing* no arranjo planar apresenta um ganho mais acentuado em relação aos métodos matriciais, como era esperado de acordo com os resultados apresentados na Figura 4.14. Nesse caso os métodos ALS-PARATUCK2 e ALS-Tucker4 com LS-KRF apresentam desempenhos semelhantes.

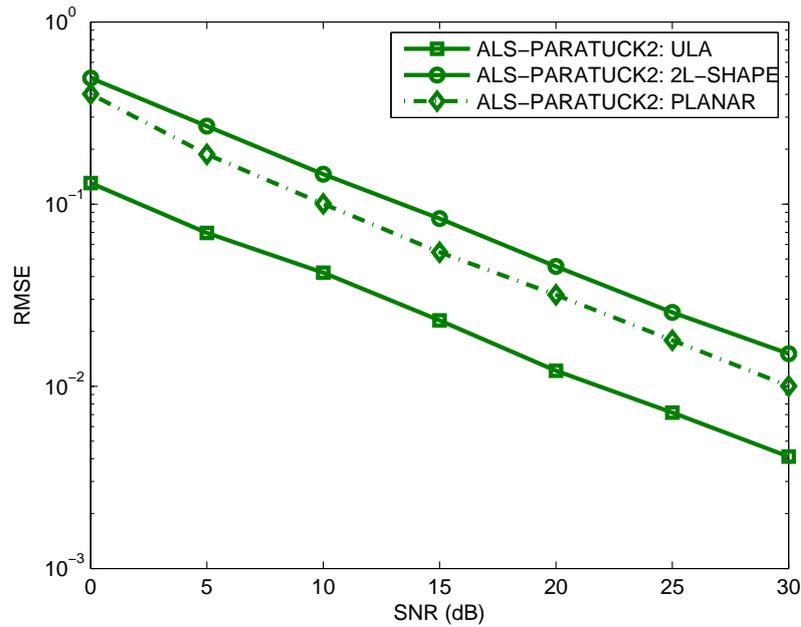


**Figura 4.18:** Tempo médio de processamento *versus* SNR para  $M = 2, P = 25, T_s = 50$  e  $N_x$  e  $N_z$  iguais a 4 e configuração Planar no receptor.

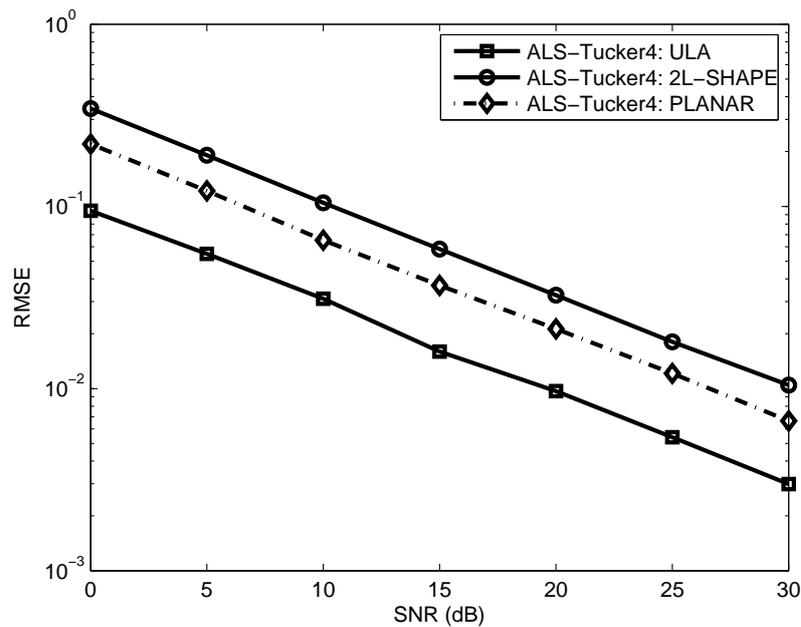
Na Figura 4.18 a análise do desempenho é realizada em termos do tempo médio de processamento em segundos em função da SNR. Note que apenas o ALS-Tucker4 adicionado do LS-KRF apresenta um tempo de processamento semelhante ao MUSIC quando um passo  $\delta = 0.5$  é utilizado. Os demais métodos propostos apresentam menor duração mesmo em comparação ao MUSIC para  $\delta = 1$ . Em comparação aos métodos baseados na decomposição PARAFAC e ao ESPRIT os métodos tensoriais propostos possuem um custo computacional superior. Entretanto, de acordo com os resultados apresentados nas Figuras 4.16 e 4.17, mesmo com o aumento do tempo médio de processamento estes métodos tornam-se mais atraentes em relação ao erro de estimação de DoA descrito pelo RMSE, principalmente nos casos em que a técnica *Spatial Smoothing* é utilizada na recepção dos sinais. Nesses casos, o algoritmo LS-KRF foi adicionado aos algoritmos propostos.

#### 4.4 Comparação entre ULA, 2L-Shape e Planar

Os resultados apresentados nesta seção descrevem a análise do desempenho entre os métodos propostos considerando arranjos ULA, 2L-Shape e Planar. Em todos os casos o cenário analisado é formado por  $M = 2, P = 20, T_s = 50$  e arranjos compostos pelo mesmo número de sensores,

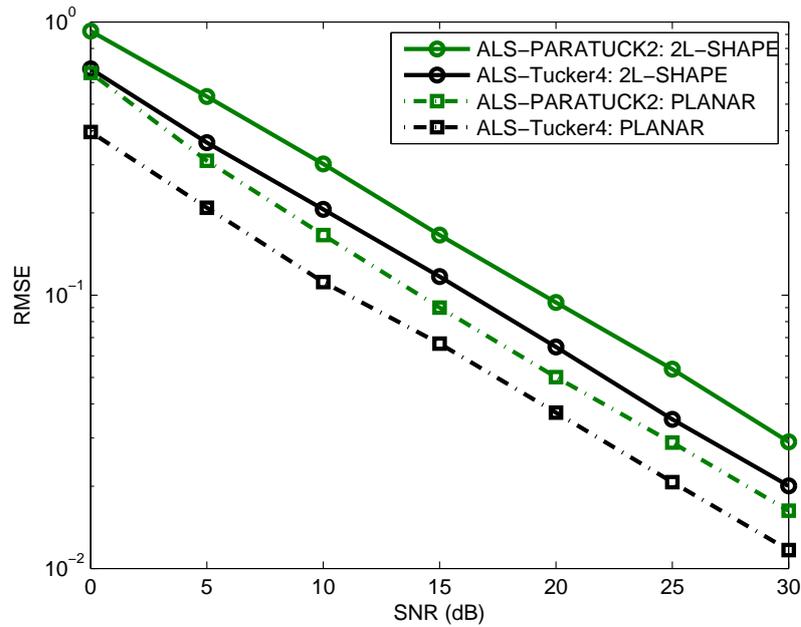


**Figura 4.19:** RMSE (elevação) *versus* SNR para  $M = 2, P = 20, T_s = 50$  e arranjos compostos por 9 sensores e utilizando a decomposição PARATUCK2.



**Figura 4.20:** RMSE (elevação) *versus* SNR para  $M = 2, P = 20, T_s = 50$  e arranjos compostos por 9 sensores e utilizando a decomposição Tucker4.

ou seja,  $N = 9$ . Os pares  $(\alpha_m, \beta_m)$  são dados por  $(35.6^\circ, 55.2^\circ)$  e  $(61.7^\circ, 72.6^\circ)$ . Nas Figuras 4.19 e 4.20 apenas estimações para os ângulos de elevação são consideradas. Note que em ambos os casos a utilização de um arranjo linear uniforme no receptor proporciona um desempenho superior em relação as configurações 2L-Shape e Planar. Entretanto, sua utilização torna-se inviável em aplicações que estimam conjuntamente os ângulos de azimute e elevação



**Figura 4.21:** RMSE (total) versus SNR para  $M = 2, P = 20$  e  $T_s = 50$  e arranjos compostos por 9 sensores no total.

são necessárias. Com relação as demais configurações, nota-se que um melhor desempenho é obtido quando um arranjo planar é utilizado.

Diferentemente dos resultados apresentados nas Figuras 4.19 e 4.20 em que apenas estimções para os ângulos de elevação das fontes foram consideradas, a Figura 4.21 descreve a análise do desempenho dos métodos propostos em termos do RMSE total, ou seja, nesse caso leva-se em consideração a estimção conjunta dos pares azimuth e elevação. Nota-se que de forma semelhante aos experimentos anteriores, a utilização do arranjo planar também torna-se mais atraente em relação ao arranjo 2L-Shape quando o ângulo de azimuth é considerado, independentemente do método utilizado na estimção da DoA das fontes.

## Conclusões e Perspectivas

Nesta dissertação o desempenho da estimação da direção de chegada dos sinais incidentes nos arranjos ULA, *2L-Shape* e Planar foi investigado. Foram propostos métodos de estimação cegos baseados em decomposições tensoriais, e algoritmos iterativos baseados em ALS foram formulados. Estes métodos foram obtidos a partir da utilização de um esquema de diversidade em blocos de tempo, provenientes da alteração da potência das fontes em blocos de dados transmitidos sucessivamente, explorando as decomposições PARATUCK2 e Tucker4 de tensores de terceira e quarta ordem obtidos a partir da covariância espacial e espaço-temporal dos dados recebidos no arranjo de sensores, respectivamente.

Com o intuito de se obter um melhor desempenho na estimação, um aumento na diversidade espacial foi alcançado com a utilização da técnica *Spatial Smoothing* no processo de recepção de sinais nos arranjos *2L-Shape* e Planar. Esta ação também proporcionou que os dados recebidos fossem modelados como decomposições PARATUCK2 e Tucker4 tornando possível, dessa forma, realizar a estimação da matriz de assinaturas espaciais das fontes utilizando os algoritmos desenvolvidos anteriormente.

Nesse caso, diferentemente da abordagem anterior, não é possível obter diretamente a estimação dos ângulos de elevação e azimute, pois nessas configurações a estrutura da matriz estimada está relacionada com o produto de Khatri-Rao entre as matrizes de interesse que possuem informação à respeito da DoA das fontes. Em virtude disso, a fatoração do produto de Khatri-Rao obtida utilizando o algoritmo LS-KRF baseado na decomposição em valores singulares foi incorporada aos algoritmos propostos, tornando possível dessa forma, realizar a estimação conjunta da matriz de assinaturas espaciais e de defasagem das quais são extraídas estimações para os ângulos de elevação e azimute das fontes.

Os métodos propostos neste trabalho apresentam características atraentes pois podem ser aplicados quando o canal de propagação, geometria do arranjo e estrutura da covariância das fontes forem desconhecidos. Entretanto, uma característica marcante desses métodos pode ser observada no desempenho alcançado quando a matriz de covariância das fontes é desconhecida e não-diagonal, o que ocorre quando se trabalha com covariâncias de amostras reais calculadas a partir de um número finito de *snapshots*, cenário em que a utilização do método proposto em [10] não se aplica. Os resultados obtidos mostram que em comparação aos métodos matriciais clássicos (MUSIC e ESPRIT) os ganhos obtidos são mais evidenciados quando a técnica *Spatial Smoothing* é utilizada.

Como perspectivas para trabalhos futuros, sugere-se a abordagem dos seguintes casos:

- ▶ Utilização de fontes fixas que proporcionam um conhecimento a priori da DoA, e conseqüentemente das colunas da matriz de assinaturas espaciais relacionadas à estas fontes, melhorando a inicialização dos algoritmos propostos;
- ▶ Extensão da análise de desempenho em cenários mais realistas, ou seja, considerando os ganhos de recepção obtidos para diferentes valores de azimute e elevação;
- ▶ Extensão da abordagem tensorial para o caso de arranjos cúbicos 3D na recepção;
- ▶ Utilização das diferentes configurações de arranjos em sistemas MIMO, de forma que a estrutura algébrica tensorial poderá ser explorada tanto no receptor como no transmissor.

# Referências Bibliográficas

- [1] Y. Rong, “Blind signal spatial signature estimation using PARAFAC model,” Master’s thesis, Faculty of Electrical Engineering, Gerhard Mercator University, 2002.
- [2] H. Krim and M. Viberg, “Two decades of array signal processing research,” *IEEE Signal Processing Magazine*, vol. 13, no. 4, pp. 67–94, July 1996.
- [3] J. P. C. L. da Costa, “Desenvolvimento de modelos, técnicas e aplicações para arranjo de sensores,” Master’s thesis, Universidade de Brasília, 2006.
- [4] N. D. Sidiropoulos, R. Bro, and G. B. Giannakis, “Parallel factor analysis in sensor array processing,” *IEEE Transactions on Signal Processing*, vol. 48, no. 8, pp. 2377–2388, august 2000.
- [5] J. P. C. L. da Costa, F. Roemer, M. Weis, and M. Haardt, “Robust R-D parameter estimation via closed-form PARAFAC,” *International ITG Workshop on Smart Antennas (WSA 2010)*, 2010.
- [6] A. L. F. de Almeida, G. Favier, and J. C. M. Mota, “PARAFAC-based unified tensor modeling for wireless communication systems with application to blind multiuser equalization,” *Signal Processing*, 2007.
- [7] —, “Space-time spreading-multiplexing for MIMO wireless communication systems using the PARATUCK-2 tensor model,” *Signal Processing*, vol. 89, 2009.
- [8] K. Liu, J. P. C. L. da Costa, A. L. F. de Almeida, and H. C. So, “A closed form solution to semi-blind joint symbol and channel estimation in MIMO-OFDM systems,” *IEEE International Conference on Signal Processing, Communication and Computing (ICSPCC)*, 2012.

- [9] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *Society for Industrial Applied Mathematics*, vol. 51, no. 3, pp. 455–500, 2009.
- [10] Y. Rong, S. A. Vorobyov, A. B. Gershman, and N. D. Sidiropoulos, "Blind spatial signature estimation via time-varying user power loading and parallel factor analysis," *IEEE Transactions on Signal Processing*, vol. 53, no. 5, pp. 1697–1710, May 2005.
- [11] J. E. F. del Rio and M. F. Catedra-Perez, "The matrix pencil method for two-dimensional direction of arrival estimation employing an l-shaped array," *IEEE Transactions on Antennas and Propagation*, vol. 45, no. 11, pp. 1693–1694, November 1997.
- [12] W. G. Diab and H. M. Elkamchouchi, "A deterministic approach for 2D-DOA estimation based on a v-shaped array and a virtual array concept," *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, 2008.
- [13] Z. Ye, Y. Zhang, and X. Xu, "Two-dimensional direction of arrival estimation in the presence of uncorrelated and coherent signals," *IET Signal Processing*, vol. 3, no. 5, pp. 416–429, September 2009.
- [14] N. Tayem and H. M. Kwon, "L-shape 2-dimensional arrival angle estimation with propagator method," *IEEE Transactions on Antennas and Propagation*, vol. 53, no. 5, pp. 1622–1630, may 2005.
- [15] F. Harabi, H. Changuel, and A. Gharsallah, "Estimation of 2-D direction of arrival with an extended correlation matrix," *4th Workshop on Positioning, Navigation and Communication WPNC-07*, 2007.
- [16] D. Liu and J. Liang, "L-shaped array-based 2-D DOA estimation using parallel factor analysis," *8th World Congress on Intelligent Control and Automation*, 2010.
- [17] R. Muhamed, "Direction of arrival estimation using antenna arrays," Ph.D. dissertation, Faculty of the Virginia Polytechnic Institute and State University, 1996.
- [18] S. Shirvani-Moghaddam and S. Almasi-Monfared, "A comprehensive performance study of narrowband DOA estimation algorithms," *International Journal on Communications Antenna and Propagation IRECAP-2011*, vol. 1, no. 4, 2011.

- [19] L. C. Godara, "Applications of antenna arrays to mobile communications. II. Beam-forming and direction-of-arrival considerations," *Proceedings of the IEEE*, vol. 85, no. 8, pp. 1195–1245, August 1997.
- [20] J. Capon, "High-resolution frequency-wavenumber spectrum-analysis," *Proceedings of the IEEE*, vol. 57, no. 8, pp. 1408–1418, August 1969.
- [21] R. O. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, pp. 276–280, March 1986.
- [22] R. O. Schmidt and R. E. Franks, "Multiple source DF signal processing: An experimental system," *IEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, pp. 281–290, March 1986.
- [23] S. Sekizawa, "Estimation of arrival directions using MUSIC algorithm with a planar array," *IEEE International Conference on Universal Personal Communications*, vol. 1, 1998.
- [24] M. Xia, "New method of effective array for 2-D direction-of-arrival estimation," *International Journal of Innovative Computing, Information and Control*, vol. 2, no. 6, pp. 1391–1397, December 2006.
- [25] A. J. Barabell, "Improving the resolution performance of eigenstructure-based direction-finding algorithms," *IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP-83*, vol. 8, 1983.
- [26] R. Roy and T. Kailath, "ESPRIT - estimation of signal parameters via rotational invariance techniques," *IEEE Transactions on Acoustic Speech and Signal Processing*, vol. 37, no. 7, pp. 984–995, July 1989.
- [27] H. Changuel, F. Harabi, and A. Gharsallah, "2-1-shape two-dimensional arrival angle estimation with a classical subspace algorithm," *IEEE International Symposium on Industrial Eletronics*, vol. 1, 2006.
- [28] M. Haardt, M. D. Zoltowski, C. P. Mathews, and J. Ramos, *ESPRIT and Closed-Form 2-D Angle Estimation with Planar Arrays*, C. P. LLC, Ed. CRC Press LLC, 1999.
- [29] T. Kuroda, N. Kikuma, and N. Inagaki, "DOA estimation of multipath waves using 2D-ESPRIT with triangular antenna array," *Proceedings of the International Symposium on Antennas and Propagation*, 2000.

- [30] M. F. Khan and M. Tufail, "Computationally efficient 2D beamspace matrix pencil method for direction of arrival estimation," *Digital Signal Processing*, vol. 20, no. 6, pp. 1526–1534, 2010.
- [31] M. Haardt, M. D. Zoltowski, C. P. Mathews, and J. A. Nossék, "2D unitary ESPRIT for efficient 2D parameter estimation," *IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP-95*, vol. 3, 1995.
- [32] R. Bro, "Multi-way analysis in the food industry: models, algorithms and applications," Ph.D. dissertation, Royal veterinary and agricultural university, 1998.
- [33] A. L. F. de Almeida, "Tensor modeling and signal processing for wireless communication systems," Ph.D. dissertation, Université de Nice-Sophia Antipolis - École Doctorale STIC, 2007.
- [34] A. Cichocki, R. Zdunek, A. H. Phan, and S.-I. Amari, *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-way Data Analysis and Blind Source Separation*, Wiley, Ed. John Wiley and Sons, Ltd, 2009.
- [35] L. D. Lathauwer, B. D. Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM J. Matrix Anal. Appl.*, vol. 21, no. 4, pp. 1253–1278, 2000.
- [36] R. L. de Lacerda Neto, "Receptores MIMO baseados em algoritmo de decomposição PARAFAC," Master's thesis, Universidade Federal do Ceará, 2005.
- [37] L. D. Lathauwer and B. D. Moor, "From matrix to tensor: multilinear algebra and signal processing," *4th IMA Int. Conf. on Mathematics in Signal Processing*, 1996.
- [38] F. L. Hitchcock, "The expression of a tensor or a polyadic as a sum of products," *J. Math. Phys.*, vol. 6, 1927.
- [39] R. Cattell, "Parallel proportional profiles and other principles for determining the choice of factors by rotation," *Psychometrika*, vol. 9, no. 4, pp. 267–283, 1944.
- [40] L. R. Tucker, "Some mathematical notes on three-mode factor analysis," *Psychometrika*, vol. 31, no. 3, pp. 279–311, 1966.

- [41] R. A. Harshman, "Foundations of the PARAFAC procedure models and conditions for an explanatory multimodal factor analysis," *UCLA Working Papers in Phonetics*, vol. 16, 1970.
- [42] J. D. Carroll and J.-J. Chang, "Analysis of individual differences in multidimensional scaling via an n-way generalization of Eckart-Young decomposition," *Psychometrika*, vol. 35, no. 3, pp. 283–319, September 1970.
- [43] R. A. Harshman, "Determination and proof of minimum uniqueness conditions for PARAFAC," *UCLA Working Papers in Phonetics*, 1972.
- [44] J. B. Kruskal, "Three-way arrays: rank and uniqueness of trilinear decompositions with application to arithmetic complexity and statistics," *Linear Algebra and Applications*, vol. 18, no. 2, pp. 95–138, 1977.
- [45] N. D. Sidiropoulos, G. B. Giannakis, and R. Bro, "Blind PARAFAC receivers for DS-CDMA systems," *IEEE Transactions on Signal Processing*, vol. 48, no. 3, pp. 810–823, march 2000.
- [46] N. D. Sidiropoulos and R. Bro, "On the uniqueness of multilinear decomposition of N-way arrays," *Journal of Chemometrics*, 2000.
- [47] A. Kapteyn, H. Neudecker, and T. Wansbeek, "An approach to n-mode components analysis," *Psychometrika*, vol. 51, no. 2, pp. 269–275, june 1986.
- [48] J. M. F. ten Berge and A. K. Smilde, "Non-triviality and identification of a constrained Tucker3 analysis," *Journal of Chemometrics*, vol. 16, 2002.
- [49] R. A. Harshman, "PARAFAC2: Mathematical and technical notes," *UCLA Working Papers in Phonetics*, vol. 22, 1972.
- [50] J. D. Carrol, S. Pruzansky, and J. B. Kruskal, "CANDELINC: A general approach to multidimensional analysis of many-way arrays with linear constraints on parameters," *Psychometrika*, vol. 45, no. 1, pp. 3–24, march 1980.
- [51] R. A. Harshman and M. E. Lundy, "Uniqueness proof for a family of models sharing features of tuckers three-mode factor analysis and PARAFAC/CANDECOMP," *Psychometrika*, vol. 61, no. 1, pp. 133–154, 1996.

- [52] A. Kibangou and G. Favier, "Blind joint identification and equalization of Wiener-Hammerstein communication channels using PARATUCK-2 tensor decomposition," *15th European Signal Processing Conference*, 2007.
- [53] G. Favier, M. N. Costa, A. L. F. de Almeida, and J. M. T. Romano, "Tensor space-time coding for MIMO wireless communication systems," *Signal Processing*, vol. 92, 2012.
- [54] S. Miron, X. Guo, and D. Brie, *Vector sensor array processing for polarized sources using a quadrilinear representation of the data*. INTECH, 2010, ch. 2, pp. 19–36.
- [55] X. Gong, Y. Xu, and Z. Liu, "On the equivalence of Tensor-MUSIC and Matrix-MUSIC," *7th International Symposium on Antennas, Propagation and EM Theory - ISAPE06*, 2006.
- [56] S. Miron, N. L. Bihan, and J. I. Mars, "Vector-sensor MUSIC for polarized seismic sources localization," *EURASIP Journal on Applied Signal Processing*, 2005.
- [57] F. Roemer and M. Haardt, "Multidimensional unitary Tensor-ESPRIT for non-circular sources," *IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP09*, 2009.
- [58] M. Haardt and F. Roemer, "Enhancements of unitary ESPRIT for non-circular sources," *IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP04*, vol. 2, 2004.
- [59] M. Haardt, F. Roemer, and G. D. Galdo, "Higher-order SVD-based subspace estimation to improve the parameter estimation accuracy in multidimensional harmonic retrieval problems," *IEEE Transactions on Signal Processing*, vol. 56, no. 7, pp. 3198–3213, July 2008.
- [60] R. Bro, N. D. Sidiropoulos, and G. B. Giannakis, "A fast least squares algorithm for separating trilinear mixtures," *Int. Workshop on Independent Component Analysis and Blind Separation*, 1999.
- [61] X. Guo, S. Miron, and D. Brie, "Identifiability of the PARAFAC model for polarized source mixture on a vector sensor array," *IEEE International Conference on Acoustic, Speech and Signal Processing - ICASSP08*, 2008.
- [62] —, "Three-way array analysis on polarized signals for direction-finding and blind source separation," *IAR*, 2007.

- [63] J. S. Paiva and A. L. F. De Almeida, “Modelagem tensorial para estimação de parâmetros em arranjos l-shape de antenas vetoriais,” *SBRT 2013 - XXXI Simpósio Brasileiro de Telecomunicações*, 2013.
- [64] S. Miron, X. Guo, and D. Brie, “DOA estimation for polarized sources on a vector-sensor array by PARAFAC decomposition of the fourth-order covariance tensor,” *16th European Signal Processing Conference - EUSIPCO08*, 2008.
- [65] S. Liu and G. Trenkler, “Hadamard, Khatri-Rao, Kronecker and other matrix products,” *International Journal of Information and Systems Sciences*, vol. 4, no. 1, pp. 160–177, 2008.
- [66] A. Smilde, R. Bro, and P. Geladi, *Multi-way Analysis: applications in the chemical sciences*, Wiley, Ed. John Wiley and Sons, Ltd, 2004.
- [67] F. Roemer and M. Haardt, “Tensor-based channel estimation and iterative refinements for two-way relaying with multiple antennas and spatial reuse,” *IEEE Transactions on Signal Processing*, vol. 58, no. 11, pp. 5720–5735, november 2010.
- [68] F. Roemer, “Advanced algebraic concepts for efficient multi-channel signal processing,” Ph.D. dissertation, Ilmenau University of Technology, 2013.